CPU

VRAM
Descubra os seus segredos

QUALIDADE CARTA NA MTA

PACLAND
O jogo

ÁGUIA INFORMÁTICA LTDA.
AV. N. S. DE COPACABANA, 605/804
COPACABANA
22040 — RIO DE JANEIRO — RJ
TELEFONE: PABX (021) 235-3541
FAX: (021) 235-3541
TELEX: 21.21717 KPUR BR

DIRETOR RESPONSÁVEL
GONÇALO R. F. MURTEIRA

DIRETOR COMERCIAL
JOSÉ IDEMAR A. NASCIMENTO

ADMINISTRAÇÃO
MÁRCIA HAUCH

PUBLICIDADE
MÁRCIA COUTINHO

ASSINATURAS
MONICA VICENTE

JORNALISTA RESPONSÁVEL
DOLAR TANUS
REGISTRO 430-RS

COLABORADORES
PAULO MARQUES FIGUEIRA
SERGIO GUY PINHEIRO ELIAS
PAULO ROBERTO PINHEIRO ELIAS
BRUNO MARRUT
JULIO VELLOSO
SÉRGIO DURIC CALHEIROS
GUILHERME A. L. DA SILVA
ANDRÉ L. A. SANTOS

REVISÃO DE TEXTO
LAURA MARIA PINTO CERSOSIMO

CAPA
JOSÉ AGUILERA

ARTE ANÚNCIOS
JORGE W. NACARI

ARTE FINAL
ADMIR DE CARVALHO
CLEBER DE JESUS PEREIRA

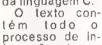
PRODUÇÃO GRÁFICA
GILSON DE S. FERNANDES
JOÃO ALVES MARTINS

COMPOSIÇÃO, MONTAGEM E FOTOLITO
GGM — GAZETA MERCANTIL
TELEFONE: 253-7893

IMPRESSÃO
GRÁFICA BRASILIANA

DISTRIBUIÇÃO
FERNANDO CHINAGLIA DISTRIBUIDORA

Integrando Clipper com linguagem C

Este livro destina-se a usuários que desejam integrar os recursos de CLIPPER com a utilização da linguagem C.

O texto contém todo o processo de integração das rotinas escritas em C, com programas escritos em CLIPPER.

O autor apresenta também dicas e rotinas em C já prontas, a título de exemplos.

Coleções de jogos

A Energy preparou para os usuários que gostam de jogos a mais completa relação para os micros da linha MSX. São mais de 70 coleções com 6 jogos, manual, gravação profissional e garantia de 60 dias.

Os usuários de MSX 2.0 também encontram na Energy mais de 20 jogos especiais além de aplicativos.

O endereço da Energy é caixa postal 18686 — 04699 — São Paulo — SP — Tel.: 011-240-1383.

Programando em Lingugem C

Kenneth Pugh

Um dos livros mais simples e práticos para se entender e programar em linguagem C.

Um verdadeiro ABC de programação em C. Pela sua simplicidade é de fácil entendimento para iniciantes e ao mesmo tempo contém informações preciosas para programadores experientes.

O autor facilita, ainda, a transição das linguagens Basic, Fortran, Cobol, PL/I e Pascal para a lingugem C.

Discovery Informática

A Discovery assinou, recentemente, contrato com Alexandre Cruz, um dos melhores programadores nacionais.

O primeiro lançamento será um programa para a confecção de páginas gráficas, em tamanho poster. O programa foi totalmente desenvolvido em Pascal e apresenta excelente desempenho e rapidez.

Os usuários dos micros DD Plus também já poderão encontrar na Discovery uma série de jogos adaptados e que carregam perfeitamente.

Maxi Dados

O Maxi Dados é um dos mais práticos bancos de dados do mercado.

Todas as suas opções são acessadas por menu e seu formato em cartucho possibilita usar mais de 50 Kb da memória.

É possível especificar o tipo de dado que cada campo irá receber, evitando erros de digitação. A ordenação é feita em memória, através de uma rotina em linguagem de máquina, sendo extremamente rápida.

Os arquivos podem ser gravados em fita ou disco e a impressão de relatórios ou etiquetas também pode ser feita, podendo se definir quais os registros deverão ser impressos, bem como os campos e formatos desejados.

Maiores informações poderão ser obtidas na Orionsoft, através do telefone 011-881-9204.

Manual Autocad para Desenho Mecânico

Donald D. Volsinet

O autor apresenta os comandos e as características do AUTOCAD de maneira prática, lógica e clara, tanto para usuários leigos quanto para os avançados.

O texto é disposto em uma seqüência progressiva que proporcionará ao usuário um aprendizado melhor em curto periodo de tempo.

Os conceitos de CAD apresentados no livro estão aplicados a um desenho prático e real gerado por CAD.

Nashua lança Disquetes com vacina antivírus de computador

O software tem capacidade de reconhecer 42 lipos principais de virus e muitos subtipos encontrados em PC e microcomputadores, além de examlnar cada programa carregado na memória bloqueando a execução dos programas contaminados.

Com esta inovação, que vai revolucionar o mercado, a Nashua pretende satisfazer às necessidades dos usuários de seus disquetes com uma arma eficiente contra o virus do computador, que se tornou o inimigo número um dos operadores de micros e PC.

Segundo o Gerente de Divisão de Disquetes da Nashua, J. J. Browne, "oferecer um brinde de muita utilidade ao usuário sem nenhum custo adicional, representa uma estratégia de marketing consonante com a alta qualidade do produlo e com a garantia por tempo ilimitado oferecido pela Nashua — única no mercado de disquetes — confirma a total confiança que temos em fazer tal lançamento".
Em praticamente um ano, a Nashua conquistou 30% do mercado de disquetes com os modelos 5 1/4, 5 1/4 H. D. e 3 1/2 que são comercializados através de distribuidores, revendedores, magazines e lojas especializadas em todo o pais.

A Nashua iniciou suas atividades diretas no Brasil em 1971. Sua fábrica toi inaugurada em 1981 em Campo Grande, Rio de Janeiro, e a empresa tem hoje cerca de 500 funcionários e filiais no Rio de Janeiro, São Paulo, Porto Alegre e Recite.

Durante os meses de fevereiro e março, a Nashua do Brasil estará distribuindo gratuitamente — junto com a caixa do modelo 5 1/4 — disquetes com programa especial antivlrus de computador.

Datamak Eletrônica

A Datamak reúne, em seu laboratório, os mais modernos softwares para diagnóstico, instrumentos de análise e aferição, bem como peças para back-up e técnicos altamente qualificados para lhe proporcionar maior segurança e contiabilidade no atendimento de equipamentos nacionais e estrangeiros.

O telefone da Datamak é 021-233-4796.

EVS Informática

Os paulistanos já podem contar com mais uma Softhouse, a EVS Informática que comercializa jogos, aplicativos, Hardware e tudo mais que um usuário de MSX necessita.

A EVS fica na Rua Aleixo Jorge 171 — Pompeia — São Paulo — SP — Tel.: 011-872-3466

DA ELETRÔNICA PARA A INFORMÁTICA

Minhas atividades na área de informática começaram há 10 anos, quando terminei o curso de Eletrônica na Escola Técnica Federal e precisava estagiar, durante 6 meses, em algum lugar. Por força do destino, acabei na Gradiente, como auxiliar técnico. Tenho, hoje, um carinho muito grande por esta empresa e nunca dispenso um elogio para os seus produtos, que são de ótima qualidade, tanto por fora quanto por dentro.

A informática veio através do estágio complementar. Desta vez, haviam me jogado para o CPD da Prefeitura de Manaus. Lá foi minha escala: de estagiário passei para conferente/perfurador, para operador (IBM SYSTEM/3), para programador (RPGII já extinto, COBOL), e, finalmente, para analista de sistema, trabalhando com um IBM 4331. Por ser usuário de máquina IBM, tive o privilégio de fazer, na IBM-Rio, todos os cursos que formam um programador em analista. A esta altura, a eletrônica, para mim, era hobby. Comecei a me apaixonar pela informática, talvez porque sempre fui apaixonado pelo desconhecido.

DO COMPUTADOR DE GRANDE PORTE PARA O MSX

No início de 85, resolvi comprar um micro para usar em casa sem nenhuma pretensão: adquiri o CP-400 da Prológica. Percebi que, apesar de ser o que eu era, mal sabia ligar o micro. Notava a grande diferença entre dois mundos: o Micro e o Main-Frame. Aprendi, por conta, o BASIC e, no final de '85, vendi meu TRS-COLOR e comprei o EXPERT da GRADIENTE. Ainda no CPD da Prefeitura, depois de passado como programalista pela SANYO, SOUZA ARNAUD, etc., recebi uma chamada do então Prefeito de Manaus, Dr. Manoel Ribeiro, para desenvolver programas particulares num micro que havia ganho. Adivinhe qual era o micro? Exatamente. Um MSX igual ao que eu tinha em casa. O Prefeito duvidou do micro e perguntou-me se um PC não seria melhor. Disse, naquele instante, que, o quê nós tínhamos era suficiente, e iniciei meus trabalhos. Pedi licença do CPD para atender o Prefeito e, a partir daí, me inclinei para os micros, deixando de lado o ICCF, VSAM, ISAM no lugar do BASIC-MSX, ASSEMBLER Z-80, MS-DOS, UNIX, etc...

CONTROLADORA DE DRIVE

Em 86, o primeiro grande obstáculo que encontrei era a falta da controladora do drive. A Gradiente, com o lançamento do EXPERT, passava por momentos difíceis de burocracia em relação a SEI e a SUFRAMA (Órgão de Superintendência da Zona Franca de Manaus), mas nós precisávamos urgentemente de uma controladora para iniciar os trabalhos. Foi quando surgiu um pequeno raio de esperança vinda do nordeste, chamado MICROSOL. Em Fortaleza, fiz diversos contatos com a diretoria da Microsol, trazendo a Manaus a primeira drive/controladora, nº série 860003 (5 1/4 FS). Naquela época, a Microsol ou a importação eram a única saída. A Gradiente, que fizera previsão de lançamentos de novos periféricos, estava sendo pressionado por todos os lados, e pela SHARP, por debaixo dos panos, pelo lançamento ilegal do seu EXPERT ESPERTO. A Microsol, pelo seu pioneirismo, coragem e determinação, merece todos os elogios de nossa parte, mas os projetistas daquela empresa foram infelizes em desenvolver uma controladora que funciona a partir de portas de I/O. Estas portas foram criadas para uma compatibilização virtual com o CP/M, mas que, segundo o manual HARDWARE SPECIFICATIONS release 5.0 — pág. 31 — de 14 de novembro de 1985 da MICROSOFT, os endereços de I/O D0-D7 não podem ser utilizados, porque o PADRÃO MSX prevê o uso de múltiplos controladoras de drive. Seguindo mais adiante, dizia: "Portanto, recomendamos o uso de endereços na memória".

Hoje, os clones da Microsol se espalham por todo o Brasil, causando um grande estrago na produção do software e na ignorância das pessoas. As controladoras da SHARP (baseado no protótipo da SONY), LEOPARD, GRADIENTE e a do ADEMIR são as únicas que se salvam deste mar de TPX, TAX, XXX etc. O pau deita e rola em cima da controladora da SHARP, onde na coluna do leitor de uma determinada revista, o leitor desabafa e indaga por que os programas PROKIT, DIGITAL BOOK, NEMESIS ESQUARTEJADO, ROBOCOP, etc., não funcionam na sua controladora que custou tão caro.

DATASET ONTEM

Às vezes, sinto-me um tanto impotente por morar tão longe do Brasil, onde tudo demora a chegar e é tão difícil de sair, onde até o horário é atrasado. Mas foi esta impotência que me deu ânimo para mudar esta situação, inverter o quadro e foi exatamente isto que eu fiz, após terminada a missão com o Sr. Prefeito. Em dez/87, na realização da 1ª Feira de Informática de Manaus, faltando apenas 3 dias e sabendo do boicote da GRADIENTE, foi que eu criei simbolicamente a DATASET INFORMÁTICA LTDA. Conseguindo 1/4 do espaço num stand emprestado, coloquei ali, naqueles dias, a minha decisão de lutar pela sigla MSX, que me conduzia para o mundo dos micros. Fomos a única segmento, diante da VERBATIM, PRODAM, CCE, DISMAC, SCOPUS, MICROTEC, etc., a representar o padrão MSX no mercado brasileiro. Naqueles dias, através de um RS-232 da SHARP, fizemos a primeira ligação MSX-PC através de modem. Sucesso tão grande que decidimos criar um CBBS local. As novidades também eram: o cartão de 80 colunas e a expan

são de 64 Kbytes da SHARP. A Gradiente nos cedeu 3 Experts c/monitores monocromáticos. O stand ficou bonito e foi um verdadeiro sucesso e as imagens deste episódio eu guardo com carinho, em casa, numa fita VHS. A feira foi como um furacão e nós não tínhamos estrutura para tanto sucesso. Todos queriam ir à DATASET comprar expansão, cartão de 80 colunas, RS-232, que eu havia conseguido a título de demonstração. Esperamos as coisas se acalmarem e, depois de 1 ano de batalha, fizemos finalmente a abertura legal da firma com instalação física própria, localizada no centro; reinstalação da rede elétrica para os micros; telefone, fax, ar condicionados, pintura novo, móveis, etc. Nossas metas:

a) assessoria e prestação de serviços;
b) desenvolvimento de programas p/ PC/MSX;
c) serviços de bureau em PC/MSX;
d) profissionalizar os MSX, mostrar a potência deste micro para os profissionais liberais, que sempre foi tratado como um video game de luxo, por culpa do marketing da GRADIENTE.

DATASET HOJE

Hoje, a DATASET ainda não tem 1 ano, mas já tem seu espaço definido em Manaus. Atualmente, ainda estou no CPD da Prefeitura, atuando mais como um administrador do que como um técnico, tendo liberdade de ir quando e à hora que quiser ou quando precisarem. Tenho um sócio, amigo daquela época do /3 da IBM, que se chama Arnaldo, engenheiro e programador. Apesar da sociedade, a prestação de serviços é independente em tudo. Somos 2 profissionais com as mesmas características. Devido ao meu colega ser engenheiro, nos inclinamos um pouco para a construção civil. Temos desenvolvido em CLIPPER um orçamento de obras integrado c/cronograma, controle de custos, contas a pagar, folha de pagamento, contabilidade padrão, estoque, etc. As empresas que atualmente estão com seus contratos (6 meses) ativos são: SERVAZ S/A SANEAMENTO E DRAGAGEM (Matriz em S. Paulo, onde nós implantamos o orçamento de obra), CONSTRUTORA CAPITAL LTDA., CONSTRUTORA SETENTRIONAL E REDE DE LOJAS INFANTIS PINGO DE GENTE. No campo MSX, desenvolvemos um controle bancário multiusuário que eu batizei de CBMU (cuja distribuição está sendo negociado com o Sr. MARCOS do MPO/SP), controle de estoque p/loja c/até 6 filiais, orçamento de obras baseado no projeto que desenvolvemos no PC, controle de patrimônio e um controle odontológico. Este último, desenvolvido durante mais de 6 meses, tomando por base informações de dois profissionais na área: Dr. C. V. Padilha (Odontólogo) e Dr. Sérgio (Professor de Odontologia na Universidade do Amazonas). Este sistema, agora, passa por uma reforma geral, porque está sendo convertido para MSX2, aproveitando-se dos recursos gráficos da máquina.

Todos estes programas foram desenvolvidos especificamente para atender um determinado tipo de cliente e agora estão passando por uma reprogramação generalizada com instalador inteligente. Em programação: SISTEMA DO ADVOGADO, que está sendo desenvolvido juntamente com dois advogados, meu irmão e o sócio dele. Em forma de bureau atendemos o SINDICATO DA CONSTRUÇÃO CIVIL DO AMAZONAS, no qual temos desenvolvido um sistema chamado C.U.B., que calcula o índice do custo unitário básico, em padrões da construção civil: H1, H4, H8 e H12. O sistema processa informações de até 10 empresas e emite mensalmente diversos relatórios estatísticos, que são enviados a Brasília, para a Fundação Getúlio Vargas, através do Sindicato local, para que seja calculado o C.U.B. de todo o Brasil. E isto é feito num MSX! Talvez, todos os outros Estados brasileiros se utilizem de um PC ou algo mais exagerado.

A venda de software importado fica a cargo da minha esposa, já que eu e meu sócio passamos a maior parte do tempo fora (em campo) prestando serviço e desenvolvendo programas PC, na própria empresa que nos contrata. No final das tardes até à noite, sábados, domingos e feriados nos encontramos na DATASET (quartel general) e mergulhamos no MSX.

DATASET AMANHÃ

A DATASET se projetou também no hardware. Vendemos, durante algum tempo, Disk-Drives de 3 1/2", disquetes MAXELL 3 1/2, impressoras, Disk cleaning 3 1/2", colocação de drives c/ controladora no EXPERT, etc. Todos estes materiais eram comprados no ZFM, dos importadores. Decidimos ser importadores para baratear o custo e ter livre escolha. Para isto, nos recadastramos junto à SUFRAMA, para adquirir cota de importação para o ano que vem. Por certo influência, é possível que consigamos uma cota de até 80 mil dólares para importação. Desde então, venho tentando contato com exportadores, através de amigos meus, para adquirir catálogos de MSX/PERIFÉRICOS para importação e venda no Brasil, através da ZFM. Imagine todos nós podermos ter em casa um MSX2+ com nota fiscal, ou uma expansão de 512Kb memory mapper, ou uma controladora de disco rígido, ou um MSX ligado a um monitor MULTSYNC II, da Nec, e tudo isso legalizado! E é possível, porque, vindo a Manaus, você tem direito de comprar e levar até $1.400 incluindo produtos de informática, sem equivalência nacional, como é o caso do MSX2 ou MSX2+. Atualmente, nas lojas da ZFM, são vendidos, livremente, para quem quiser, MAC PLUS,

PS/2 modelos 30/40, LAPTOPs, impressora à LASER, EPSON FX-1050, PANASONIC KX, monitores SANSUNG e MULTSYNC e diversos periféricos. Mas para o MSX não há nada, exceto os drives de 3 1/2 dos PCs que podem ser utilizados no MSX. Eis o problema: a maioria das importações vêm dos Estados Unidos e Panamá, e, através de diversos contatos com exportadores que conheço, recebi resposta negativa quanto à existência de MSX nesses países. A dificuldade é tanta que, para comprar o nosso MSX2+ no Japão, através de um amigo que para lá viajou, foi preciso que ele confiasse o dinheiro a um japonês para ir comprar na loja. O vendedor se negava a vender ao meu amigo, dizendo: — Only japaneses. Hoje, o micro está aqui ligado a um Telefunken AVM Stéreo com RGB de fábrica, funcionando a todo vapor.

BOAS-NOVAS

Temos um laço de amizade muito grande com o diretor-residente e técnicos da Gradiente e pessoal da Suframa. Posso adiantar alguns dados sobre a Gradiente que ainda, talvez, não sejam de conhecimento público e que são curiosos.

1º) A SUFRAMA liberou guia de importação para 50.000 drives de 3 1/2", mas ela só importou inicialmente 10.000.

2º) Ao montar a linha piloto do DD-PLUS, a maioria dos drives, aleatoriamente, davam mensagens de I/O Error. Os projetistas haviam cometido um engano na controladora: um capacitor no lugar de um trimpot, que não poderia ser reajustado. Ocorreu atraso de lançamento devido a este fato.

3º) A Gradiente, apesar de ter comprado a Telefunken, a partir de janeiro/90 abandonará este nome, passando os televisores a terem a marca GRADIENTE.

4º) Os novos EXPERTs vêm com a RAM no slot 3-3, como os importados. Mas a maioria (90%) dos boot's dos joguinhos não fazem procura de RAM em slot secundária. Conclusão: CRASH!!!

5º) Os drives de 3 1/2", embutidos no DD-PLUS, são os famosos E-Y DATA, os mesmos usados nos micros PS/2 da IBM.

6º) Para baratear o custo dos micros, a Gradiente importou o 'chipão' econômico da Toshiba, utilizado nos modelos HX20-HX22. Isso deve trazer um sério transtorno para o Ademir.

7º) A Gradiente comprou a licença do logotipo MSX2 + ROM + SUB-ROM, da MITSUBISHI (?), para os próximos 10 anos.

8º) Já há uma programação de datas para o lançamento do MSX2, da Gradiente: Maio/90 protótipo, Setembro/90 linha piloto, Novembro/90 em produção. Os projetistas convenceram com um simples argumento: tiramos da linha o Cartão de 80 cols, economizando placas, gabinetes plásticos, componentes, etc..., e o colocamos dentro da CPU. O atual cartão de 80 cols., da Gradiente, traz o coração do MSX2: o chip V-9938, da YAMAHA.

9º) A coisa parece séria mesmo. A Gradiente enviou ao Japão alguns técnicos da GRATEC, e eles retornaram trazendo diversos modelos MSX2, periféricos e manuais técnicos da ASCII japoness, em inglês, sobre o padrão MSX2. Inclusive, um dos manuais, o MSX2 TECHNICAL DATA BOOK de 400 pags, eu xeroquei e mandei de presente para o Ademir.

10º) A última informação que obtive através do pessoal da GRATEC, é a melhor de todas: a ASCII já está com o MSX3 (Terceira Geração) no forno. A PANASONIC desenvolveu um chip reconfigurável ou chip de 2 caras, de 64 bits, que tomará o lugar do veterano Z 80. (a mesma coisa aconteceu com o processador de vídeo, o V9938 MSX2, da, YAMAHA, que tem dentro de si, o antigo TMS-9128 MSX1). A velocidade de processamento do MSX3, em carga máxima, atingirá a mesma velocidade de um AT-286, da IBM. A compatibilidade, a nível de programa, entre um PC e um MSX se tornará realidade! Previsão de lançamento: Novembro de 1990.

Quero completar dizendo que todas as informações acima citadas são extra oficiais e que só o futuro poderá dizer se são VERDADEIRAS. Como vivemos no Brasil, tudo é possível.

ENTENDENDO A VRAM

PAULO MARQUES FIGUEIRA

Vamos campreender, agora, um pauca da estrutura da memária de vídeo (VRAM) dos micras MSX, a fim de aprendermos alguns truques muita interessantes.

Nossa MSX (o da versão 1) passui uma memária de vídea de 16 Kb. Esta memória é independente da RAM narmal que está disponível para os programas. Sabemos que a nasso MSX passui quatra mados de vídea, sendo a screen 0 e 1 para as modos de texta e screen 2 e 3 para os gráficas. Observe a mapa da VRAM que segue abaixa na parte da screen 0:

MAPA DA VRAM — MSX 1.0

END	SCREEN 0	SCREEN 1	SCREEN 2	SCREEN3
0	NOMES 0	PADRÕES 7	PADRÕES 12	PADRÕES 17
959				
2947				
2048	PADRÕES 2			NOMES 15
4095				
6144		NOMES 5	NOMES 10	
6911				
6912		ATRSPRIT 8	ATRSPRIT 13	ATRSPRIT 18
7039				
8192		CORES 6	CORES 11	
8223				
14335				
14336		PADRÕES SPRITES 9	PADRÕES SPRITES 14	PADRÕES SPRITES 19
16383				

Sabemas que se trata da tela de texta em 40 calunas e podemas observar duas tabelas na VRAM:

— Tabela de nomes área de vídea real. Digamos que esta seja coma uma janela para a seu vídea. Tuda a que cair nessa área surgirá na tela. Coma temos 24 linhas e 40 calunas e precisamos de um byte para cada caracter, temos 24 x 40 = 960 bytes. Par exempla, para calacar a letra "A" na segunda linha e décima caluna, então devemos fazer: VPOKE 2*40+10,65, ande:
VPOKE — calaca uma infarmação na memória de vídea
2*40+10 — calcula a endereço da VRAM (linha 2, caluna 10)
65 — código para a letra "A"

— Tabela de padrães nesta tabela temas a farmato de cada letra que o micra pode usar. Para cada caracter usamos 8 bytes para indicar o seu farmato, que está em binária, de farma que a bit 1 representa a panta acesa na tela e a bit em 0 representa o ponta apagada.

Digite o programa abaixa e observe:

```
10 SCREEN 0
20 FOR A = 2048 TO 4095
30 PRINT RIGHT$(("00000000"+BIN$ (VPEEK (A))),8)
40 NEXT A
```

O que nas interessa aqui são apenas estas duas áreas de memária. O resta está vazia (na screen 0). Aí pademas calacar várias outras telas e até mesma gerar imagens quadro a quadra, como veremas mais adiante.

No Basic MSX existe uma variável interna chamado BASE. Coma uma variável indexada, ela guarda o endereça da VRAM onde estãa as tabelas que nas interessam. Par exempla, para saber qual a endereça da tabela de caracteres (padrões) na screen 0 faça:
PRINT BASE (2)
O pragrama abaixa, feita para a screen 1, criará uma nava tabela de caracteres na seu micro, porém não vai alterar a original, fazendo cam que você fique cam dais tipas de letras:

```
10 SCREENO
20 FOR I = 0 TO 2047: W = VPEEK (I)
30 VPOKE I+2048, W OR (W/2) + (W/4)
40 NEXT
```

Você não notará diferença de imediata, pois a nava tabela de padrões fai criada em autra área da VRAM. Assim, para tracar de caracteres digite: BASE (7) = 2048.

Com issa, você mudau o endereça da tabela de padrões para 2048, ande está a nava tabela. Para valtar aa narmal, digite BASE (7) = 0. Perceba as várias recursas que isto pade nos oferecer.

Atente, agora, para as variáveis do sistema na tabela abaixo:

CONTEUDO	SCREEN0 END.	SCREEN1 END.	SCREEN2 END.	SCREEN3 END.
END.TABELA DE NOMES	F3B3	F3BD	F3C7	F3D1
TABELA DE CORES		F3BF	F3C9	
PADRÕES DE CARACTERES	F3B7	F3C1	F3CB	F3D5
ATRIBUTOS DOS SPRITES		F3C3	F3CD	F3D7
PADRÕES DOS SPRITES		F3C5	F3CF	F3D9

Quando mudamos os endereços da VRAM através da variável BASE(), na verdade, estamos apenas, mudando estes endereços de memória onde a VDP irá ler as informações e mostrar no vídeo. Mas para que possamos realmente trabalhar noutra área da VRAM, é necessária a comando SCREEN, que fará uma mudança completo.

Logo, devemos modificar estes endereços utilizando BASE () e depois SCREEN. Podemos, com isso, ter várias telas montadas na VRAM e trocá-las com uma velocidade incrivelmente alta. Há apenas uma limitação quanto à SCREEN 2, já que esta ocupa quase todos os 16Kb da memória de vídeo.

O programa da listagem 1 é um bom exemplo deste processo. Nele, vemos um cubo desenhado em três quadros na SCREEN 3. Estes quadros vão sendo trocados um por um, dando a ilusão de movimento. Note que, para desenharmos o cubo em outra área de memória, é necessário o uso dos comandos BASE e SCREEN, mas, para apenas exibirmos outras páginas, basta usar BASE.

A listagem 2 nos traz mais duas procedures para os programadores em Pascal. Você poderá incluí-las às que foram publicadas em CPU número 7 página 25. Apenas note uma pequena mudança feita na procedure screen. A procedure VPAGE faz uma mudança em até treze páginas em screen e uso a procedure BASE. Para VPAGE devemos informar o número da página seguida do valor 0 para apenas mostrar a página ou 1 para realmente efetuar a mudança, quando, então, a nova página será limpa.

Usando este recurso na SCREEN 0, podemos armazenar até treze telas cheias (cada uma possui apenas 960 bytes) na VRAM e exibi-los mudando o endereço de BASE (0), criando rápidas mudanças de vídeo em seus programas.

Por fim, não se esqueça de levar em conta que:

— O endereço da tabela de nomes deve ser múltiplo de 1024;

— O endereço da tabela de caracteres deve ser múltiplo de 2048;

— O endereço da tabela de cores deve ser múltiplo de 64;

— O endereço da tabela de atributos dos sprites deve ser múltiplo de 128;

— O endereço da tabela de padrões dos sprites deve ser múltiplo de 2048.

Esta é uma exigência do processador de vídeo e todos estes endereços podem, também, começar do zero.

Espero, com isso, ter mostrado um pouco de mais um dos várias truques que se pode fazer com o VDP truques estes que, muitas vezes, estão bem à nosso frente dentro de um aplicativo ou um jogo, e nem sequer são notados. Até à próxima.

```
5   '*********** FIM ****************
6   '*  GERANDO EFEITOS NA SCREEN 3   *
10 '*Programa exemplo:(USO DA VRAM)  *
20 '*AUTOR: PAULO MARQUES FIGUEIRA   *
30 '*DATA:25/08/89                   *
40 '*            REVISTA   CPU       *
41 '*********************************
50 OPEN"GRP:"AS#1
60 X=70:COLOR 12,1:BASE(17)=0:BASE(15)=
2048:SCREEN5
70 LINE(130,10)-(40,30):LINE-(40,160):L
INE-(130,180):LINE-(220,160):LINE-(220,
30):LINE-(130,10):LINE-(130,180):PAINT(
120,179),2,12:PAINT(140,170),7,12
71 LINE(0,150)-(40,150),12:LINE(220,150
)-(255,150):PAINT(0,190),13,12
80 GOSUB230
90 COLOR 12,1:BASE(17)=4096:BASE(15)=81
92:SCREEN3
100 LINE(40,30)-(40,160):LINE-(80,180):
LINE-(220,160):LINE-(220,30):LINE-(80,1
0):LINE-(80,180):LINE(80,10)-(40,30):PA
INT(70,160),2,12:PAINT(140,170),7,12
101 LINE(0,150)-(40,150),12:LINE(220,15
0)-(255,150):PAINT(0,190),13,12
110 GOSUB 230
120 COLOR 12,1:BASE(17)=10240:BASE(15)=
12288:SCREEN3
130 LINE(40,30)-(40,160):LINE-(180,180)
:LINE-(220,160):LINE-(220,30):LINE-(180
,10):LINE-(40,30):LINE(180,10)-(180,180
):PAINT(185,170),2,12:PAINT(140,170),7,
12
131 LINE(0,150)-(40,150),12:LINE(220,15
0)-(255,150):PAINT(0,190),13,12
140 GOSUB 230
150 BASE(17)=0:BASE(15)=2048
160 FORT=1TOX:NEXT
170 BASE(17)=4096:BASE(15)=8192
180 FORT=1TOX:NEXT
190 BASE(17)=10240:BASE(15)=12288
200 FORT=1TOX:NEXT
210 I$=INKEY$:IFI$<>""THENCOLOR 15:END
220 GOTO 150
230 COLOR14:X$="LEIA CPU":FORV=1TOLEN(X
$):PSET(V*27-5,100),POINT(V*27-5,100):P
RINT#1,MID$(X$,V,1):NEXT:COLOR12:RETURN
```

```
(*+-----------------------------------+*)
(*|PROCEDURES DE PAGINACAO DE VIDEO|*)
(*|EM TURBO PASCAL - COMPLEMENTO DO|*)
(*|PACOTE GRAFICO PUBLICADO EM  CPU|*)
(*|NUMERO 7 - POR PAULO M. FIGUEIRA|*)
(*+-----------------------------------+*)

PROCEDURE SCREEN(X:BYTE);
(*              |       |   *)
(*comando------+       |   *)
(*modo da tela---------+   *)

BEGIN
      IF (X>3) THEN
         BEGIN
               INLINE($F7/$00/$6C/$00);
               WRITELN('ERRO DE SCREEN');
               HALT;
      END;
      CASE X OF
      0 : INLINE($F7/$00/$6C/$00);
      1 : INLINE($F7/$00/$6F/$00);
      2 : INLINE($F7/$00/$72/$00);
      3 : INLINE($F7/$00/$75/$00);
      END;
END;

PROCEDURE BASE(Z:BYTE ; X:INTEGER);
(*              |   |            | *)
(*COMANDO---+   |            | *)
(*NUM DA BASE---+            | *)
(*ENDERECO-------------------+ *)

VAR Y:STRING[3];
    W:STRING[6];

BEGIN
      STR(Z,Y); STR(X,W);
      COMP:='('+Y+')'+CHR($EF)+W+CHR(0);
      INLINE($21/$00/$A0/$DD/$21/
             $5A/$7B/$CD/$00/$C0);
END;

PROCEDURE VPAGE(X,    Y:BYTE );
(*           |  |     | *)
(*COMANDO----+  |     | *)
(*NUM DA PAGINA-+     | *)
(*1=mudar 0=mostrar---+ *)

VAR SCREN:   BYTE ABSOLUTE $FCAF;
    NUMES:   INTEGER;

BEGIN
     IF SCREN>0 THEN


        BEGIN
              SCREEN(0);
              WRITELN('SCREEN ERRADA!');
              HALT;
        END;
     IF (X<0) OR (X>12) THEN
        BEGIN
              SCREEN(0);
              WRITELN('PAGINA MUITO ALTA');
              HALT;
        END;
     IF (Y>1) THEN
        BEGIN
              SCREEN(0);
              WRITELN('ERRO DE PARAMETRO');
              HALT;
        END;
     IF X>=2 THEN X:=X+2;
     NOMES:=1024*X;
     BASE(0,NOMES);
     IF Y=1 THEN SCREEN(0);
END;
```

DISCOVERY INFORMÁTICA

MSX
AMIGA
TK90
PC

TECNOLOGIA EM SOFTWARE NACIONAL.

MSX FLOW CHART

por VITOR HUGO P. COSTA

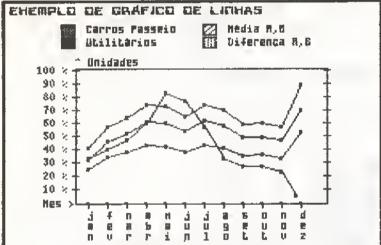
Finalmente o primeiro programa de Gráficos comerciais e estatísticos profissional para MSX. Totalmente desenvolvido em linguagem Pascal, o programa possui recursos que o tornam em muitos casos, superior aos melhores programas para 16 bits.

Algumas características do programa:

- 7 tipos de gráficos (bi e tri-dimensionais).
- funções matemáticas definíveis.
- recursividade nas funções matemáticas.
- arquivamento de dados e formatos.
- entrada de dados até 100 ítens.
- impressão em escala de cinzas.
- arquivamento de telas em vários formatos.
- aplicação comercial e científica.

- ACOMPANHA MANUAL COMPLETO.
- DISPONÍVEL EM 5 1/4 E 3 1/2 POLS.
- GARANTIA E SUPORTE PERMANENTES.
- PROGRAMA 100 % NACIONAL.
- REGISTRADO NO I.N.P.I.

PREÇO : NCZ$ 1.100,00

DESKTOP PUBLISHING

DESKTOP SURFACES ➜ SUPERFÍCIES PARA ACABAMENTO EM DESKTOP. NCZ$ 270,00

BORDERS #2 ➜ NOVO CONJUNTO DE MOLDURAS COM FORMATOS INÉDITOS. NCZ$ 270,00

SUPER LETTERS #2 ➜ SENSACIONAL CONJUNTO DE LETRAS EM FORMATO SHAPE. NCZ$ 320,00

ART PACK #2 ➜ FIGURAS INCRÍVEIS E TOTALMENTE INÉDITAS PARA DESKTOP. NCZ$ 270,00

LETTERS #2 ➜ NOVO CONJUNTO DE FONTES INÉDITAS E ACENTUADAS. NCZ$ 230,00

Estas páginas foram criadas com os seguintes produtos: Super Letters #2, Desktop Surfaces, MSX Flow Chart, letters #2 e Graphos Pro.

DESKTOP PRESENTATION

COLOR SHAPES ➜ SENSACIONAIS SHAPES COLORIDOS PARA ABERTURAS EM VIDEO CASSETE OU TELAS DE JOGOS. NCZ$ 330,00

COLOR SURFACES ➜ SUPERFICÍES COLORIDAS PARA ACABAMENTO EM VINHETAS PARA VIDEO CASSETE, TELAS DE JOGOS OU QUALQUER OUTRO TRABALHO QUE NECESSITE GRANDE APELO VISUAL. ACOMPANHA MANUAL COM DICAS PARA A CONFECÇÃO DE BONITAS VINHETAS PARA VIDEO CASSETE. NCZ$ 340,00

AMIGA, O COMPUTADOR DA DÉCADA ➜ FITA DE VIDEO MOSTRANDO TODA A POTENCIALIDADE DESTE MICRO MARAVILHOSO. QUALIDADE PROFISSIONAL. FORMATO VHS. NCZ$ 780,00

Coleções Desktop Publ. & Pres. requerem Graphos 3

LEITURA E GRAVAÇÃO DA VRAM

MÁRCIO MACHADO MOURA
Nemesis Informática

Embora já existam rotinas prontas no BIOS, referentes à leitura e gravação da VRAM, as mesmas não apresentam a performance, em velocidade, desejada em muitas aplicações, onde o tratamento de vídeo necessita de alto desempenho. Veremos no artigo deste mês, como interagem os equipamentos padrão MSX com o vídeo, pois a compreensão deste mecanismo nos possibilitará o detalhamento do uso direto das portas de entrada e saída, associadas aos microprocessadores, envolvidos no gerenciamento do vídeo: O Z80, e o VDP.

MEMÓRIA DE VÍDEO

Embora seja fato que o periférico vídeo seja um periférico de saída, temos uma situação diferente, em se tratando de equipamentos de tecnologia mais atual, como no caso dos microcomputadores padrão MSX. Neste tipo de arquitetura, temos um microprocessador dedicado apenas ao tratamento de vídeo, que, através de um programa especial, administra uma memória RAM (de 16 Kb nos modelos MSX 1.0), diretamente associada ao vídeo. Esta associação faz com que o vídeo funcione como uma espécie de espelho desta memória, sendo impossível a gravação em determinadas áreas de memória sem o reflexo direto desta gravação no vídeo. Sendo assim, é possível ler o que aparece no vídeo, pois esta aparição é apenas uma codificação dos dados contidos em uma memória RAM possibilitando a recuperação destes dados, que, aos nossos olhos, formam imagens no vídeo.

Um detalhe importantíssimo na compreensão da memória de vídeo, denominada de VRAM, refere se à sua divisão em áreas, que cumprem funções dedicadas à formação de imagens no vídeo:

1) ÁREA DE FORMAÇÃO DE CARACTERES;
2) ÁREA DE APARIÇÃO DE CARACTERES;
3) ÁREA DE CORES DOS CARACTERES;
4) ÁREA DE FORMAÇÃO DE SPRITES;
5) ÁREA DE LOCALIZAÇÃO DE SPRITES.

Considerando o objetivo de comunicação com o usuário que o periférico vídeo requer, temos como área base, a ÁREA DE FORMAÇÃO DE CARACTERES, onde existe o desenho de todos os símbolos e letras da tabela ASCII. Como veremos no decorrer do artigo, isto não impede que tenhamos apenas desenhos sem a presença de nenhum caracter na tela.

A ÁREA DE APARIÇÃO DE CARACTERES é realmente a parte mais associada ao vídeo, pois, quando gravamos um código ASCII nesta área, a imagem dele será lida, da ÁREA de FORMAÇÃO DE CARACTERES, e o desenho do caracter aparecerá imediatamente na tela. É bom observar que a relação de posição entre a área e o vídeo é linear. Isto significa que não existem linhas e colunas na geração de imagens, sendo apenas simuladas por software nas várias aplicações que necessitam deste tipo de referencial. Na realidade, temos que considerar que o byte seguinte ao último byte de uma linha, é o primeiro byte da linha abaixo, e assim sucessivamente, até o fim da tela. Por isso, a gravação sequencial na ÁREA DE APARIÇÃO DE CARACTERES, obriga a aparição dos desenhos dos caracteres no vídeo, pulando de linha (na imagem vista na tela), sempre que a quantidade de bytes de uma linha se esgota.

A terceira área refere-se às cores que serão usadas no desenho dos caracteres. Como são usados duas cores para cada byte de desenho, temos aquilo que é conhecido como cor de frente e cor de fundo. Na realidade, o que temos é uma cor, que será usada para "pintar" os zeros do byte (cor de fundo), e uma outra cor, que será usada para "pintar" os uns do byte (cor de frente).

As duas últimas áreas, correspondentes aos SPRITES, serão apenas citados no detalhamento dos modos da SCREEN, pois sua análise será realizada em outro artigo, por envolverem maiores detalhes que serão discutidos na oportunidade.

MODOS DE SCREEN

Existem 4 modos de utilizarmos a VRAM, sendo administradas pelo VDP todas as peculiaridades de cada modo. Cada um deles se utiliza de uma técnica na geração de imagens da mais simples (SCREEN 0) até a mais sofisticada (SCREEN 2). Por não ter uma utilidade muito prática, em comparação aos demais modos, não analisaremos o último modo, correspondente à SCREEN 3. Passemos, então, à análise isolada de cada modo, descrevendo a maneira de uso de cada área da VRAM.

SCREEN 0

Modo que serve basicamente para criação de textos, sendo o modo que mais possibilita caracteres por linhas de imagem. Ele permite a criação de telas textos, com até 40 colunas por 24 linhas. O que nos dá, para ÁREA DE APARIÇÃO DE CARACTERES, um espaço de 960 bytes (40x24). A forma que o VDP encontrou para isto, foi usar apenas 6 pontos de largura para cada caracter mostrado no vídeo, o que nada atrapalha no uso de textos, pois todas as figuras da tabela ASCII envolvidas com texto (letras, sinais ortográficos, etc), foram desenhados com apenas 5 pontos de largura.

A tabela de caracteres onde existe uma figura, para todos os códigos de 0 a 255, ocupa um espaço de 2 Kb, pois separa 8 bytes na formação de cada caracter, possibilitando criar imagens em uma matriz de 64 pontos (8 linhas de bytes, por 8 colunas de bits). Esta tabela, com o desenho de todos os caracteres, reside na ROM da BIOS, do endereço 1BBFH até 23BEH, e é carregado na VRAM ao se escolher um modo de SCREEN (0 ou 1), no respectivo endereço, associado à ÁREA DE FORMAÇÃO DE CARACTERES do modo selecionado.

Por ser um modo mais associado à textos, não existe para SCREEN 0 a área de cores e as áreas de SPRITES, ficando a VRAM praticamente vazia, onde apenas 3008 bytes (2048 + 960) são utilizados, podendo se usar o resto do espaço para o que se possa imaginar, até mesmo dados do programa, sem que apareçam sujeiras no vídeo ou que origine outro problema qualquer. Vejamos, então, as posições das áreas da VRAM no modo 0 de SCREEN:

ÁREA DE APARIÇÃO DE CARACTERES	0000H — 03BFH	(960 BYTES)
VAZIO	03C0H — 07FFH	(1088 BYTES)
ÁREA DE FORMAÇÃO DE CARACTERES	0800H — 0FFFH	(2048 BYTES)
VAZIO	1000H — 3FFFH	(12288 BYTES)

Obs.: Os valores dos endereços de início e fim de áreas estão expressos em hexadecimal. O tamanho das áreas, colocados entre parênteses, estão expressos em decimal.

SCREEN 1

Modo intermediário entre texto e gráfico, que permite todos os recursos do modo anterior, mas que já possibilita maior flexibilidade, podendo utilizar área de cores e alguns recursos gráficos (como SPRITES). Trabalha com toda a imagem do caracter (8 pontos), o que implica em uma perda de 8 colunas de vídeo, formando textos de até 32 colunas por 24 linhas, o que nos dá um total de 768 bytes para ÁREA DE APARIÇÃO DE CARACTERES.

A ÁREA DE FORMAÇÃO DE CARACTERES funciona de modo análogo ao modo 0, também sendo carregada da ROM do BIOS.

A grande novidade neste modo reside na possibilidade de utilizarmos cores no desenho dos caracteres, de uma forma, porém, um tanto quanto curiosa. A ÁREA DE CORES DOS CARACTERES ocupa um espaço de apenas 32 bytes da VRAM, o que nos deixa "pintar", com cores diferentes de frente e fundo, grupos sequenciais de 8 caracteres (256 / 32 = 88). Cada byte indica a cor de 8 caracteres seguidos, sendo o primeiro byte para os primeiros 8 caracteres (código de 0 a 7, da tabela ASCII), o segundo para os códigos de 8 a 15, e assim sucessivamente. A maneira de guardar duas cores (frente e fundo) em apenas um byte é simples: Os 4 bits da esquerda suportam 16 códigos de cor, de 0000 a 1111, e indicam a cor de frente; os 4 bits da direita funcionam de forma análoga, e representam a cor de fundo. Gostaria, apenas, de ressaltar, neste assunto de cores, que a cor 0, conhecida como transparente, de incolor não tem nada. Representa, apenas, a cor da borda, ou seja, quando algo tem cor 0 em um equipamento MSX, significa que este algo assumirá a mesma cor da borda. Vejamos, então, o mapeamento da VRAM para o modo 1 de SCREEN:

ÁREA DE FORMAÇÃO DE CARACTERES	0000H — 07FFH	(2048 BYTES)
VAZIO	0800H — 17FFH	(4096 BYTES)
ÁREA DE APARIÇÃO DE CARACTERES .	1800H — 1AFFH	(768 BYTES)
ÁREA DE LOCALIZAÇÃO DE SPRITES ..	1B00H — 1B7FH	(128 BYTES)
VAZIO	1B80H — 1FFFH	(1152 BYTES)
ÁREA DE CORES DOS CARACTERES	2000H — 201FH	(32 BYTES)
VAZIO	2020H — 37FFH	(6112 BYTES)
ÁREA DE FORMAÇÃO DE SPRITES	3800H — 3FFFH	(2048 BYTES)

Obs.: Como dito anteriormente, embora não analisadas as áreas referentes aos SPRITES foram citadas quanto às suas posições na VRAM.

SCREEN 2

Modo gráfico não muito recomendado paro uso de textos. Mais dedicado oo uso de desenhos. Neste modo, é comum se dizer que possibilita à criação de imagens com 512 colunas por 192 linhas, só que de pontos, no lugar de caracteres. Este enfoque é apenas referencial, para ratificar a idéia de associação com gráficos, que tem este modo.

Na verdade, o modo 2 de SCREEN administra 3 telas, com 3 áreas de formação, aparição e cores dos carcteres: a primeira tela corresponde às 8 primeiras linhas do video; a segunda vai da linha 9 até a linha 16; e a última da linha 17 a 24.

No momento que este modo é estabelecido, as três ÁREAS DE APARIÇÃO DE CARACTERES são preenchidas com os códigos de 0 a 255, e três ÁREAS DE FORMAÇÃO DE CARACTERES são preenchidas com bytes 0. Assim, o comum é criarmos imagens, mexendo nestas áreas de formação, que estão inicialmente vazios, pois como as áreas de aparição já estão preenchidas, o envio de qualquer byte para as áreas de formação fará o byte aparecer como parte da imagem do código N correspondente à ÁREA DE APARIÇÃO DE CARACTERES. Notem que, por termos três telas com áreas distintas para todas elas, poderíamos, se quisessemos, escrever textos neste modo, usar até três alfabetos, sendo, cada um correspondente a uma das três telas.

A ÁREA DE CORES DOS CARACTERES também é muito maior. Podemos, na realidade, ter uma cor para cada byte (frente e fundo), escrita na ÁREA DE FORMAÇÃO DE CARACTERES. Portanto, as duas áreas são da mesma tamanho.

Mesmo neste modo, que usa quase toda a VRAM, ainda temos 1152 bytes livres, que não são usados para nada. Isto se dá, pois as memórias possuem capacidade múltipla de base 2, e o que mais chega perto da quantidade necessária ao uso do maior modo (SCREEN 2) é a de 16Kb. Por isso, a sobra das 1152 bytes. O mapo dos áreas correspondente ao modo 2 de SCREEN é o seguinte:

ÁREA DE FORMAÇÃO DE CARACTERES	0000H — 17FFH	(6144 BYTES)
ÁREA DE APARIÇÃO DE CARACTERES	1800H — 1AFFH	(768 BYTES)
ÁREA DE LOCALIZAÇÃO DE SPRITES	1B00H — 1B7FH	(128 BYTES)
VAZIO	1B80H — 1FFFH	(1152 BYTES)
ÁREA DE CORES DOS CARACTERES	2000H — 37FFH	(6144 BYTES)
ÁREA DE FORMAÇÃO DE SPRITES	3800H — 3FFFH	(2048 BYTES)

Obs.: As áreas foram mostradas em grupos totais das três telas. Abaixo, temos a subdivisão, por tela, de cada uma das áreas:

TELA 1:

ÁREA DE FORMAÇÃO DE CARACTERES	0000H — 07FFH	(2048 BYTES)
ÁREA DE APARIÇÃO DE CARACTERES	1800H — 18FFH	(256 BYTES)
ÁREA DE CORES DOS CARACTERES	2000H — 27FFH	(2048 BYTES)

TELA 2:

ÁREA DE FORMAÇÃO DE CARACTERES	0800H — 0FFFH	(2048 BYTES)
ÁREA DE APARIÇÃO DE CARACTERES	1900H — 19FFH	(256 BYTES)
ÁREA DE CORES DOS CARACTERES	2800H — 2FFFH	(2048 BYTES)

TELA 3:

ÁREA DE FORMAÇÃO DE CARACTERES	1000H — 17EEH	(2048 BYTES)
ÁREA DE APARIÇÃO DE CARACTERES	1A00H — 1AFFH	(256 BYTES)
ÁREA DE CORES DOS CARACTERES	3000H — 37EFH	(2048 BYTES)

CONCLUSÃO

Substitui o nome das áreas da VRAM, em comparação aos encontradas em literatura especializada sobre MSX, puramente com fins didáticos, pois os nomes geralmente utilizados, como TABELA DE NOMES, ATRIBUTOS, não significariam muita coisa para aqueles desacostumados com a manipulação da VRAM.

Talves alguns dos conceitos definidos neste artigo não tenham ficado muito claros. Porém, a única forma de compreendermos certos detalhes é colocando as em prática. No nosso próximo artigo, veremos os famosos SPRITES, onde já apresentaremos rotinas em linguagem ASSEMBLY, que nos possibilitarão, um uso muito mais claro e racional, na leitura e gravação da VRAM. Estas rotinas, apresentam alta performance, por manipularem diretamente as portas de video, libertando, assim, o uso exclusivo das rotinas da BIOS no que se refere ao envio e recuperação de dados da VRAM.

SIMULANDO QUALIDADE CARTA NA GRAFIX – MTA

LUIS GUSTAVO MILFONT PEREIRA

Assim como eu, várias pessoas que adquiriram uma impressora Grafix—MTA, tiveram alguma frustração pois, ao ler o manual que a acompanha, descobriram que ela não possui o modo QUALIDADE CARTA.

Em vista disso, criei uma rotina em assembler que permite que se imprima neste modo, que foi elaborada em um EXPERT PLUS, no Mega Assembler.

COMO O PROGRAMA FUNCIONA

O programa será colocado a partir do endereço &HC000.

Existem alguns códigos em hexadecimal, que, ao serem jogados na rotina LPTOUT da bios (&H00A5), significam um comando para a impressora. No programa, carrega-se o acumulador com 1BH. Em seguida, joga-se o valor na rotina do bios. Este valor faz com que a impressora aguarde um comando. Após, carrega-se o acumulador com 41H e, pelo mesmo procedimento anterior, chama-se a rotina LPTOUT. O código 41H é um comando especial que altera o espaçamento das linhas na impressora. O próximo byte, terá o valor do espaçamento. No caso, 00H. Então , não há espaçamento. Exatamente. E é assim que o programa funciona. Ele imprime um determinado texto várias vezes sobre o mesmo lugar, fazendo com que a escrita fique mais escura, e, conseqüentemente, mais parecida com as letras de uma máquina de escrever.

Se você quiser deixar o texto mais escuro, altere o endereço &HC028, que contém o número de vezes que a cabeçote da impressora passará pelo mesmo lugar. Se você colocar um valor maior, mais escuro o texto ficará. Se, o valor for menor, mais claro. Assim, você pode controlar a intensidade de impressão, e ajustar para que a impressão não fique "borrada".

O código 32H restaura o espaçamento natural da impressão.

Se quiser alterar os códigos de impressão, lembre-se que o comando deve vir precedido de 1BH, e que estes códigos devem ser jogados na rotina LPTOUT, como caracteres normais.

No manual da Grafix—MTA, existem tabelas de códigos de comando para a impressora (pags. 21 e 22 — "Modo Texto"), mas se você não o possui aí vão alguns códigos:

CR	— Carriage Return	— 0DH
LF	— Line Feed	— 0AH (AVANÇA LINHA)
FF	— Form Feed	— 0CH (AVANÇA FOLHA)
ESC+"C"+N		— 1BH 43H NH (COMPRIMENTO DA FOLHA)
ESC+"A"+N		— 1BH 41H NH (ESPAÇAMENTO)
ESC+"2"		— 1BH 32H (RESTAURA ESPAÇAMENTO)
SO		— 0EH (CARACTER EXPANDIDO)

Espero ter contribuído de alguma forma para os leitores desta excelente revista, possuidores de uma Grafix—MTA. Se você tiver alguma dica envie para a CPU. Compartilhe suas idéias com o público, pois quanto mais informações temos ao nosso alcance melhor.

Luiz Gustavo Milfont Pereira, 7/7/1973.
Cursos: Basic, Basic avançado, Computação Gráfica, Inglês Autodidata em Assembler.
Profissão: Estudante.

```
10 SCREEN0:KEYOFF
20 FORT=&HC000 TO &HC04F
30 READ A$:POKE T,VAL("&H"+A$)
40 NEXT
50 DEFUSR=&HC000
60 A=USR(0)
70 DATA 3E,1B,CD,A5,00,3E,41,CD
80 DATA A5,00,3E,00,CD,A5,00,16
90 DATA 00,D5,21,37,C0,7E,CD,A5
100 DATA 00,23,FE,00,C2,15,C0,3E
110 DATA 0D,CD,A5,00,D1,14,7A,FE
120 DATA 0F,C2,11,C0,3E,1B,CD,A5
130 DATA 00,3E,32,CD,A5,00,C9,4C
140 DATA 55,47,55,4D,49,50,45,20
150 DATA 53,4F,46,54,57,41,52,45
160 DATA 21,00,00,00,00,00,00,00
```

Listagem em BASIC do programa: Qualidade Carta na GRAFIX-MTA Digite a listagem se não possuir um monitor de linguagem de maquina.

Caso voce tenha um monitor, digite a listagem em LM, e grave-a com: BSAVE "QC.BIN",&HC000,&HC04F,&HC000

```
<<  Mega Assembler 1.0 >>

                    ;                  Qualidade Carta na GRAFIX-MTA.
                    ;               Por: Luiz Gustavo Milfont Pereira.
                              ORG   0C000H
  00A5              LPTOUT:EQU   00A5H
  C000 3E1B               LD    A,01BH
  C002 CDA500             CALL  LPTOUT
  C005 3E41               LD    A,041H
  C007 CDA500             CALL  LPTOUT
  C00A 3E00               LD    A,00H
  C00C CDA500             CALL  LPTOUT
  C00F 1600               LD    D,00H
  C011 D5       LOOP:     PUSH  DE
  C012 2137C0             LD    HL,TXT
  C015 7E       ROT:      LD    A,(HL)
  C016 CDA500             CALL  LPTOUT
  C019 23                 INC   HL
  C01A FE00               CP    00H
  C01C C215C0             JP    NZ,ROT
  C01F 3E0D               LD    A,0DH
  C021 CDA500             CALL  LPTOUT
  C024 D1                 POP   DE
  C025 14                 INC   D
  C026 7A                 LD    A,D
  C027 FE0F               CP    0FH
  C029 C211C0             JP    NZ,LOOP
  C02C 3E1B               LD    A,01BH
  C02E CDA500             CALL  LPTOUT
  C031 3E32               LD    A,032H
  C033 CDA500             CALL  LPTOUT
  C036 C9                 RET
  C037 4C554755 TXT:      DB    'LUGUMIPE SOFTWARE!
  C03B 4D495045
  C03F 20534F46
  C043 54574152
  C047 4521
  C049 00                 DB    0
  C04A                    END
```

Listagem em LM do programa: Qualidade-carta na GRAFIX-MTA.

Para gravar(pelo BASIC): BSAVE "QC.BIN",&HC000,&HC04F,&HC000

PROGRAMA
UTILITÁRIO

Alternativas de impressão para o MSX-WORD

SÉRGIO GUY PINHEIRO ELIAS
PAULO ROBERTO PINHEIRO ELIAS

INTRODUÇÃO

O MSX-WORD é um dos mais utilizados Processadores de Texto pelos usuários de microcomputadores da linha MSX. Este sucesso é, em grande parte, explicado pelas facilidades de manuseio que este aplicativo oferece, traduzidas pelo seu domínio através de um período de treinamento relativamente reduzido, no qual é possível produzir arquivos de texto com um acabamento bastante razoável, sem que se tenha tido alguma experiência prévia significativa.

De fato, é possível observar-se que, quando o usuário esbarra em dificuldades na utilização de um programa, a tendência é abandoná-lo, proporcionalmente ao grau dos impecilhos e obstáculos na compreensão do seu funcionamento. Por outro lado, quando os resultados esperados são obtidos a curto prazo, a empatia entre o usuário e a máquina aumenta o processo de aceitação e o desejo de avançar na aplicação dos seus recursos.

O MSX-WORD talvez tenha sido o primeiro editor de Textos capaz de falar e imprimir caracteres de língua portuguêsa de forma correta. Além disso, devido a qualidades intrínsecas ao seu funcionamento, foi o primeiro aplicativo do gênero que conhecemos adaptável às impressoras disponíveis há alguns anos, as quais não possuíam nenhuma das Tabelas de caracteres do padrão MSX,

Talvez sejam estes fatores que mencionamos acima os responsáveis pela permanência deste Editor na preferência do usuário, pois, em alguns aspectos, em níveis de exigência, digamos, mais profissionais, o programa deixa a desejar, tanto em relação aos recursos de edição, quanto aos recursos de impressão dos textos digitados.

A EVOLUÇÃO DO MSX-WORD

Ultimamente, a CIBERTRON vem se esforçando no sentido de minorar as deficiências notórias que o Editor apresenta, ainda para o usuário da série 1.0 do MSX, em ampla maioria no país, o qual, normalmente, sequer faz uso de um monitor de vídeo adequado para o funcionamento do computador.

Até a Versão 3.0 aparecer, a MSX-WORD não era capaz de aproveitar o funcionamento de um segundo drive, para arquivamento dos textos e até a Versão 1.6 não conseguia imprimir mais do que um arquivo-texto por vez. Isto sem falar na incompatibilidade da linha de texto gravada pelo programa, com outros Processadores que conseguiriam importar seus arquivos.

Quando a Versão 3.0 estava sendo fabricada, tivemos a oportunidade de apresentar alguns pontos de vista nossos aos programadores da CIBERTRON, alguns dos quais foram aceitos e implementados no programa, como por exemplo, a retirada de algumas rotinas executadas, com pouca freqüência, pelo usuário, do bloco em BASIC principal do programa e a adoção de uma única rotina de leitura e gravação de arquivos, com possibilidade de verificar a listagem do diretório e a memória livre real do disco de cada um dos drives A e B.

A primeira destas providências citadas desmontou parcialmente o bloco principal em BASIC em arquivos de overlay perfeitamente substituíveis. Isto, por si só, determinou uma enorme economia de memória na micro, facilitando sobremaneira o funcionamento do programa, sem o chamado "crash" no computador (vulgarmente, "pau no programa"). O usuário da Versão 3.0 poderá notar uma performance mais equilibrada desta Versão, sem praticamente nenhum tipo de interrupção nefasta ao seu desempenho, sem comparação, neste particular, com as Versões anteriores.

Todos os pontos relativos à acomodação do programa no computador, bem como as rotinas de gravação e leitura sofreram todos os melhorias que tinham direito. Em contrapartida, a rotina de impressão de arquivos-texto permaneceu rigorosamente a mesma, talvez porque, não tenha havido tempo hábil suficiente para que nova rotina tivesse sido implementada. Seria, como de fato é, necessário alterar a parte em linguagem de máquina do programa, para que ocorresse uma mudança radical na concepção da forma de digitação e edição de modo a acomodar mais de um tipo de impressão.

Possivelmente, uma nova Versão, ainda a caminho, deverá solucionar de vez com estes percalços. Enquanto isto não ocorre, tudo o que poderemos fazer é oferecer algumas alternativas de impressão, sem, é claro, radicais compromissos com a qualidade final, pois isto deverá ficar por conta da futura nova Versão.

AS ALTERNATIVAS

Três programas são apresentados a seguir, mostrados nas Figuras 1 a 3. Os dois primeiros trabalham externamente ao Processador, lendo diretamente no disco de trabalho onde as arquivos de texto estão gravados, para depois imprimi-los da

- os melhores cursos -

assistência técnica especializada -

- mais de 30.000 clientes -

- o maior estoque do mercado -

- mais de 2.000 programas -

a mais completa linha de periféricos -

Equipamentos · Acessórios · Periféricos

Interfaces · Drives · 80 colunas · Modem

O MAIOR SHOW ROOM DO PAÍS !!!

RUA APIACÁS, 92 – SÃO PAULO - CEP 05017 / FONE 872-0730
FILIAL: AEROPORTO-SP
RUA NHU GUASSU, 188 – TEL: (011) 241-8912

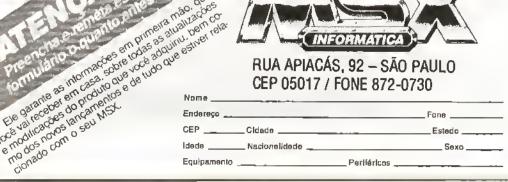
SEJA NOSSO REPRESENTANTE NA SUA CIDADE. CONSULTE-NOS.

Nome ______________________________

Endereço ______________________ Fone ________

CEP ________ Cidade ______________ Estado ________

Idade ________ Nacionalidade ______________ Sexo ________

Equipamento ______________ Periféricos ______________

maneira selecianada pelo usuário. O terceira funciana junta com o programa original, na farma de um overlay, partanta a partir de um arquivo previamente lido e instalada na memória do micro.

Os programas das Figuras 1 e 2 são bastante parecidos: obedecem à mesma rotina de leitura da disco, mas a decadificação das linhas difere nas duas Versões. O algaritmo de leitura e impressão é baseada na fato singular de que as linhas do MSX-WORD terminam sempre pela seqüência CARRIAGE RETURN-LINE FEED e que a camando LINE INPUT# é capaz de ler uma string na disco sem delimitadores (aspas, etc.), de até 254 Bytes, até que o sinal de CARRIAGE RETURN (<CR>) seja encontrado. Quando isto ocorre, o comanda pula a seqüência <CR>-<LF> e aponta para a próxima string a ser lido pelo comando LINE INPUT# seguinte (1).

No primeira Versão que fizemos deste programa (não publicada), desmontamos a linha do arquivo-texto e decodifi camos cada um dos Bytes, de maneira a identificar qual o tipo de caracter a ser impresso: se acentuado, seria tracado pelo respectivo código da impressora (no caso de incompatibilidade), se código de controle do impressora (caracteres 192 a 207). Idem, além de alguns ajustes necessários para colocar a linha do arquivo nas posições de digitação. Infelizmente, a desmontagem, decodificação e posterior remontagem lamava um tempo tão grande (afinal de contas, este processa ocorria 68 vezes por linha !), que o desempenho da rotina acabou pecando pela lentidão excessiva. Decidimos então eliminar toda o filtro de impressão e conservar apenas os ajustes necessários para a correto posicionamento das linhas do texto na formulário. Este programa está mostrado na Figura 1. Com estas providências, o programa tornou-se rápido, porém bastante limitado de virtudes. Entretanto, poderá rodar comodamente com um filtro de impressão adequado, como aquele publicado na livro "Programação Avançada em MSX" (2).

No programa da Figura 2, foi conservado, além do ajuste das linhas, uma decodificação dos códigos de comandos de impressora dispostos nos caracteres 192 a 207 mencionadas anteriormente. O leitor deverá tomar o cuidado de trocar os códigos que constam daquela rotina para os códigos efetivamente usados na adaptação do Editor à sua impressora. No caso deste programa, a sua velocidade de execução ficará prejudicada pela lentidão da rotina de decodificação, mas não chega a ser uma coisa intolerável como na primeira caso.

Em ambos os programas das Figuras 1 e 2, o usuário recebe, na tela inicial, a chance de modificar uma série de parâmetros de impressão. Cada um deles possui o seu próprio default, listados entre parênteses após cada opção. Toda vez que um determinado parâmetro for modificado, as especificações entre parênteses mudarão de acordo. A numeração das páginas poderá ser desligada ou redefinida para um valor diferente, no caso de mais de um arquivo ser impressa sobre a mesma assunto. Teoricamente, não haverá limite de leitura de qualquer arquivo no disco. Como os arquivos do MSX-WORD são limitadas pelo tamanho, para aumentá-los será necessário mergeá-los no próprio disco (comando COPY do MSX-DOS ou utilitário COPIARQ do SOLX-DOS).

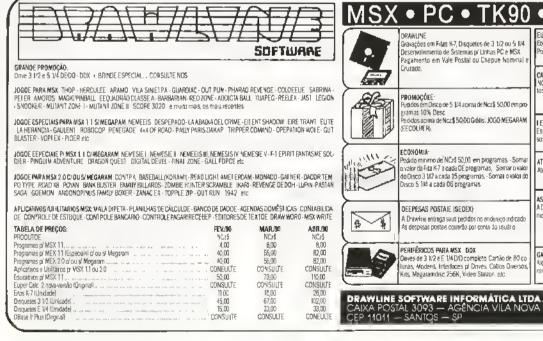
No programa da Figura 3, o mesmo deverá ser gravado no disco de trabalho, com o nome de PÁGINA. BAS ou qualquer outro nome de sua preferência, tomando o cuidado de mantê-lo igual na linha digitada a seguir, dentro do próprio MSX-WORD. As adaptações são bastante simples, bastando fazer o seguinte:

Carregue o arquivo MSXWORD3.0 do disco de trabalho. Neste bloco não deverá existir a linha 820 (em se tratando de uma cópia original do programa) e a rotina de impressão deverá começar na linha 810. Assim, digite a linha 820, com as seguintes declarações:

```
820 LOCATE 0,3:PRINT"Tecle:<1>imp/linhas<2>imp/pág.
"::O$=INPUT$(1):PRINTO$:IFO$="1"THENPRINT:GOTO830EL
SEIFO$="2"THENRUN"PAGINA.BAS"ELSE820
```

Se você tiver dúvida sobre a existência da linha 820, será sempre bom listá-la antes de digitar a nova linha. De qualquer forma, esta deverá preceder as instruções da rotina de impressão normal do MSX-WORD. Terminada a digitação, salve novamente o programa, antes de testá-lo.

Ao ser acionada, a impressão de um arquivo-texto poderá ser feita da forma habitual, pela opção <1>, ou paginada, pela <2>. Neste último caso, o arquivo-texto não deverá conter códigos de FORM FEED — CHR$ (12) — embebidos no texto, pois isto alterará a paginação imposta pela rotina. O usuário terá direito a definir o número da página ou a ausência de numeração, logo que a rotina for acionada. É obrigatório teclar <return> (ou <CR>) para numerar e N ou n para não numerar. O programa sempre apresentará a página 1 como inicial, podendo então ser redefinida para outro número. Se a numeração não for desejável, o programa pula para a opção de número de páginas por linha. O default é de 55, considerando-se o padrão de 66 entrelinhas de comprimento na página, a qual não poderá ser alterada. Por causa disso, o valor máximo para esta opção é de 60 linhas de texto. Nas opções seguintes, o usuário poderá definir as páginas inicial e final para impressão. Teclando <return> em ambos as perguntas, serão se-

lecionodos as páginas inicial e final, respectivamente. Lembre-se que, neste caso, o número das páginas será determinado pela sua relação com o volume de texto determinado pela pergunta anterior, já que não é possível saber "a priori" onde ficam estes limites no corpo do texto, a não ser que o próprio usuário tabele e delimite as linhas onde as páginas ficam.

As perguntas a seguir se referem ao número de cópias e à necessidade de pausa na impressão. Neste último item, ao contrário da rotina original, o programa realmente pára ao fim de uma página impressa, possibilitando usar folhas soltas sem problemas. Antes da impressão, o usuário tem a opção de desistir e voltar ao programa principal ou, durante a impressão propriamente dita, teclar ESCAPE com a mesma finalidade. Neste último caso, o programa abre uma tela de controle, informando cada cópia, página e linha impressas, possibilitando saber até onde a impressora irá operar automaticamente quando a rotina for interrompida. Teclando-se <STOP> ao invés de <ESC>, a impressão estanca até que a tecla seja novamente pressionada. Nenhuma rotina de interrupção de <STOP> foi colocada no programa, que pudesse interferir nesta função.

COMENTÁRIOS FINAIS

Não espere nada de excepcional no programa da Figura 3, mas podemos garantir que, pelo menos em nossos testes, ele funcionou a contento, ressalvados algumas limitações de formatação, como por exemplo, as linhas serão sempre impressas em salto simples (espaçamento 1) e o documento impresso (de tamanho fixo) não terá cabeçalho ou rodapé, como nos programas anteriores. A nosso ver, não se deve encher demais uma página de texto, sendo o default de 55 linhas/páginas suficiente para a maioria dos documentos. Todos os esforços foram feitos para tornar o programa pequeno de modo a não causar problemas na área de memória do micro, evitando qualquer possibilidade de "crash" no sistema. Esperamos que, mesmo com as limitações comentadas, ele possa ser útil a um bom número de usuários, até que nova Versão do MSX-WORD venha a público com melhores soluções.

AGRADECIMENTO PESSOAL

Desejo agradecer ao meu prezado amigo. Dr. Carlos Fernando de Carvalho Blanco, ilustríssimo Diretor de Fomento no Departamento Nacional da Produção Mineral (D.N.P.M.), no Rio de Janeiro, pelo seu generoso incentivo à minha participação na programação e divulgação de dados constantes deste artigo.

Paulo Roberto P. Elias

REFERÊNCIAS BIBLIOGRÁFICAS

1 — Elias, S.G.P. & Elias, P.R.P. Sistemas Operacionais do MSX e suas ferramentas. Editora Ciência Moderna Ltda., Rio de Janeiro, 1989.

2 — Luz, H.F. et al. Programação Avançada em MSX. Editora Aleph, São Paulo, 1967.

Figuras:

```
10 'PROGRAMA PARA IMPRIMIR
20 'ARQUIVOS-TEXTO DO MSX-WORD
30 'AUTORES:S.G.P.E./P.R.P.E.
40 'DATA: NOVEMBRO DE 1989
50 KEY OFF:LOCATE,,0:CLEAR 1000
60 ON ERROR GOTO 210
70 A$="NÃO":B$="NÃO":A=1:B=8:C=55:F1=0:F2=1:L$="SIM":R=0
80 CLS
90 PRINT"IMPRESSÃO DE ARQUIVOS-TEXTO DO MSX-WORD":LOCATE13,2:PRI
NT"Menu Principal":LOCATE1,4:PRINT"(  ) indicam estado atual das
 funções"
100 LOCATE,6:PRINT"1 - IMPRIME CABEÇALHO (";A$;")"
110 LOCATE,8:PRINT"2 - IMPRIME RODAPÉ (";B$;")"
120 LOCATE,10:PRINT"3 - NUMERAÇÃO DAS PÁGS. (";L$;")
```

JUVENTUOE E TECNOLOGIA

Cadastramos Revendedores em Todo o Brasil

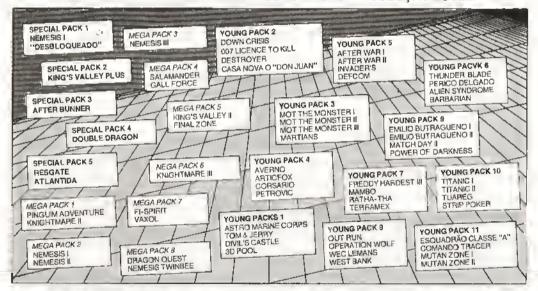
MSX TOP-SECRET

UM SUPER UTILITÁRIO TDTALMENTE INOVADOR NO MERCADO MSX, OUE COMPREENDE NADA MENOS QUE TOOAS AS QUALIOAOES OOS UTILITÁRIOS QUE JÁ FIZERAM SUCESSO NO MERCADO MSX. COM O TOP-SECRET É POSSÍVEL: ZAPPEAR UM DISQUETE, COPIAR E FORMATAR UM DISCD COM INCRÍVEL FIDELIDADE, EDITAR SPRITES BX8 E 16X16, TRANSFORMAR E ELABORAR GRÁFICOS COM O COMANDO ORAW OO BASIC, ARMAZENAR SUAS INFORMAÇÕES EM UM SUPER BANCO DE DADOS, EDITAR ALFABETOS PARA USO EM QUALQUER SCREEN, IMPRIMIR TELAS E OUTRAS SUPER FERRAMENTAS.

PALHADA CITY

PROVANDO QUE CRIATIVIDADE É DIFERENTE DE NDME COMERCIAL, A YOUNGSOFT LANÇA UM SENSACIONAL ADVENTURE QUE ALTERNA REALIOAOE E FICÇÃO, DRAMA E COMÉDIA EM UMA MISTURA HOMOGÊNEA DE MUITO SUSPENSE E BOM HUMOR. RIO DE JANEIRO 01/01/2000. E NESSA DATA HISTÓRICA QUE ARIONALDINO, FAMOSO HISTORIAOOR HOLANOÊS DESCOBRE QUE NA PACATA PALHAOA CITY EXISTEM MUITOS SERES ALIENÍGENAS E ALIENAOOS QUE ESCONDEM A MAIOR FORTUNA EM PEDRAS PRECIOSAS OO MUNOO. CONSEGUIRA ARIONALDINO ESCAPAR DE PALHADA CITY?
GRÁFICOS BELÍSSIMOS E ÓTIMA TRILHA SONORA.

Cadastramos Revendedores em Todo Brasil

YOUNG PACKS – A SOLUÇÃO JOVEM

Young Pack	: NCz$ 150,00
Mega Pack	: NCz$ 120,00
Special Pack	: NCz$ 120,00

Também temos pacotes de 5 e 10 Jogos, de aplicativos, educativos e programas em geral. Catálogo grátis.

Para fazer seu pedido
Envie carta detalhada
Para Youngsoft c/cheque
nominal a Yvone Di Santo Barboza

Prazo de entrega.
Disco = 10 Dias
Fita = 20 Dias
Garantia de 3 Meses

Temos toda linha de periféricos do mercado a preços alucinantes.

YOUNG CLUB

Associe-se ao mais ousado clube de usuários do Brasil.

Vejas as vantagens: 50% de desconto em jogos, 20% em aplicativos e 15% em revistas CPU. Um pacote de jogos por mês e um de aplicativos a cada 3 meses. Catálogo mensal com novidades da fima e do mercado em si. Concursos, Promoções, Brindes, etc.

Para fazer parte do Young Club basta enviar carta detalhada dizendo a configuração do seu equipamento, os pacotes que deseja receber e envie junto um cheque nominal e cruzado a Yvone di Santo Barboza no valor de (10) btn's (btn mensal).

Obs.: O associado só terá direito aos benefícios após o segundo pagamento. Pagamento mensal.

Youngsoft

INFORMÁTICA LTDA.

A 1ª SOFTHOUSE DE NOVA IGUAÇU

Travessa Almerinda Lucas de Azeredo 11/1211 – Centro – Nova Iguaçu – Rio de Janeiro – CEP 26150
Tele-Jovem: (021) 767-9545 / Ramal 90

```
130 LOCATE,11:PRINT"    REDEFINIÇÃO DA NUMERAÇÃO (";R;")"
140 LOCATE,13:PRINT"4 - DEFINE OFFSET DE IMPRESSÃO (";B;")"
150 LOCATE,15:PRINT"5 - DEFINE Nº DE LINHAS/PÁGINA (";C;")"
160 LOCATE,17:PRINT"6 - IMPRIME TEXTO"
170 LOCATE,19:PRINT"7 - ABANDONA O PROGRAMA"
180 LOCATE,21:PRINT"Sua opção: ";
190 C$=INPUT$(1):IF C$<"1" OR C$>"7" THEN 80 ELSE PRINT C$:FOR T
=1 TO 100:NEXT T
200 ON VAL(C$) GOTO 240,300,380,440,510,560,960
210 IF ERR<>53 THEN CLS:PRINT"FOI DETECTADO UM ERRO Nº:";ERR;"LI
NHA:";ERL:GOTO 230
220 IF ERL=610 THEN PRINT:PRINT"Arquivo não existe":CLOSE:PRINT:
PRINT"Tecle algo":T$=INPUT$(1):RESUME 560
230 CLOSE:PRINT:PRINT"Tecle algo":T$=INPUT$(1):RESUME 80
240 CLS:PRINT"OPÇÃO PARA IMPRESSÃO DE CABEÇALHO"
250 LOCATE,5:PRINT"Confirma impressão de cabeçalho (S/N) ?";:C$=
INPUT$(1)
260 IF C$="N" OR C$="n" THEN A$="NÃO":GOTO 80 ELSE IF C$="S" OR
C$="s" THEN A$="SIM":GOTO 270 ELSE 240
270 LOCATE0,7:PRINT"Digite o cabeçalho (máx.:55 caracteres)"
280 LINE INPUT D$:IF LEN(D$)>55 THEN PRINT:PRINT"Cabeçalho com m
ais de 55 caracteres":PRINT"Tecle algo" ELSE IF D$="" THEN 270 E
LSE 80
290 T$=INPUT$(1):GOTO 240
300 CLS:PRINT"OPÇÃO PARA IMPRESSÃO DE RODAPÉ"
310 LOCATE0,5:PRINT"Confirma impressão de rodapé (S/N) ?";:C$=IN
PUT$(1)
320 IF C$="N" OR C$="n" THEN B$="NÃO":F1=0:L$="NÃO":GOTO 80 ELSE
 IF C$="S" OR C$="s" THEN B$="SIM":GOTO 330 ELSE 300
330 LOCATE0,7:PRINT"Digite o rodapé (máx.:55 caracteres)"
340 LINE INPUT E$:IF LEN(E$)>55 THEN PRINT:PRINT"Rodapé com mais
 de 55 caracteres":PRINT"Tecle algo" ELSE IF E$="" THEN 300 ELSE
 360
350 T$=INPUT$(1):GOTO 300
360 LOCATE0,11:PRINT"Inclui numeração (S/N) ?";
370 C$=INPUT$(1):IF C$="N" OR C$="n" THEN F1=0:F2=0:L$="NÃO":GOT
O 80 ELSE IF C$="S" OR C$="s" THEN F1=1:F2=0:L$="SIM":GOTO 80 EL
SE 360
380 CLS:PRINT"OPÇÃO PARA NUMERAÇÃO DAS PÁGINAS"
390 LOCATE0,5:PRINT"Confirma nova paginação (S/N) ?";
```

```
400 C$=INPUT$(1):IF C$="N" OR C$="n" THEN 80 ELSE IF C$="S" OR C
$="s" THEN 410 ELSE 390
410 LOCATE0,7:PRINT"Digite 0 para não numerar páginas."
420 LOCATE0,9:INPUT"Qual o novo número da página ";A
430 IF A=0 THEN F2=0:L$="NÃO":F5=0:GOTO 80 ELSE IF A>0 THEN F2=1
:L$="SIM":R=A:F5=1:GOTO 80 ELSE 420
440 CLS:PRINT"OPÇÃO PARA OFFSET DE IMPRESSÃO"
450 LOCATE0,5:PRINT"Confirma redefinição de offset (S/N) ?";
460 C$=INPUT$(1):IF C$="N" OR C$="n" THEN 80 ELSE IF C$="S" OR C
$="s" THEN 470 ELSE 450
470 LOCATE0,7:INPUT"Qual o novo offset de impressão ";B
480 IF B>0 AND 8<17 THEN 80 ELSE IF B<0 THEN LOCATE0,7:PRINT SPA
CE$(39):GOTO 470
490 LOCATE0,11:PRINT"CUIDADO: offset alto. Redefine (S/N) ?";
500 C$=INPUT$(1):IF C$="N" OR C$="n" THEN 80 ELSE IF C$="S" OR C
$="s" THEN 470 ELSE LOCATE0,11:PRINT SPACE$(39):GOTO 490
510 CLS:PRINT"OPÇÃO PARA N' DE LINHAS/PÁGINA":LOCATE0,2:PRINT"Má
ximo aceito pelo programa: 55"
520 LOCATE0,5:PRINT"Confirma novo nº de linhas/pág. (S/N) ?";
530 C$=INPUT$(1):IF C$="N" OR C$="n" THEN 80 ELSE IF C$="S" OR C
$="s" THEN 540 ELSE 510
540 LOCATE0,7:INPUT"Qual o nº de linhas/página ";C
550 IF C<0 OR C>55 THEN LOCATE0,11:PRINT"Número de linhas/página
 ilegal":GOTO 540 ELSE 80
560 CLS:PRINT"OPÇÃO DE IMPRESSÃO DE ARQUIVOS-TEXTO"
570 LOCATE0,2:PRINT"Indique o drive (A/B) ";:F$=INPUT$(1):PRINT
F$:PRINT:PRINT
580 IF F$="A" OR F$="a" THEN FILES"A:" ELSE IF F$="B" OR F$="b"
THEN FILES "B:" ELSE 570
590 PRINT:PRINT:PRINT"Nome do arquivo:(<RETURN> sai da opção)"
600 LINE INPUT G$:IF G$="" THEN 80 ELSE G$=F$+CHR$(58)+G$
610 OPEN G$ FOR INPUT AS #1
620 CO=1:F4=0:A=I:PRINT:PRINT"Começa impressão na página (1): ";
:
630 INPUT CO:IF CO=0 THEN CO=1:A=1 ELSE IF CO=2 THEN A=2:CO=1:F4
=1 ELSE IF CO>2 THEN A=CO:CO=CO-1 ELSE IF CO<0 THEN PRINT:PRINT"
Valor ilegal":GOTO 620
640 TE=20:PRINT:PRINT"Termina na página (última): ";:
650 INPUT TE:IF TE<CO THEN TE=CO+1 ELSE TE=TE+1
660 IF F5=1 THEN PRINT:PRINT"Numeração redefinida para:";R:PRINT
"Confirma (S/N) ?"; ELSE 700
670 C$=INPUT$(1):IF C$="S" OR C$="s" THEN A=R:GOTO 710 ELSE IF C
$="N" OR C$="n" THEN 680 ELSE 660
680 PRINT:PRINT"Abandona impressão (S/N) ?;"
690 C$=INPUT$(1):IF C$="S" OR C$="s" THEN 80 ELSE IF C$="N" OR C
$="n" THEN 710 ELSE 680
700 PRINT:PRINT"Pausa entre as páginas (S/N) ?";
710 C$=INPUT$(I):IF C$="S" OR C$="s" THEN F6=1 ELSE IF C$="N" OR
 C$="n" THEN F6=0 ELSE 700
720 PRINT:PRINT:PRINT"Prepare a impressora e tecle algo";:T$=INP
UT$(1)
730 CLS:PRINT"[ESC] = sai da impressão":PRINT:PRINT"[STOP] = pau
sa/continua"
740 IF CO>1 OR F4=1 THEN GOSUB 980
750 REM INICIA IMPRESSÃO
760 IF A$="SIM" THEN LPRINT TAB(B);D$ ELSE LPRINT
```

```
770 LPRINT
780 X$=INKEY$:IF X$=CHR$(27) THEN CLOSE:GOTO 80
790 FOR I=1 TO C
800 IF EOF(1) THEN CLOSE:F3=1:GOTO 860
810 X$=INKEY$:IF X$=CHR$(27) THEN CLOSE:GOTO 80
820 LINE INPUT#1,H$
830 I$=MID$(H$,2,67)
840 LPRINT TAB(B);I$
850 NEXT I
860 K=I-1
870 FOR L=1 TO 55-K:LPRINT:NEXT L
880 IF B$="SIM" THEN LPRINT TAB(B);E$;"   ";
890 IF F1=1 THEN LPRINT"Pág.:";A:GOTO 910
900 IF F2=1 THEN LPRINT TAB(39);A
910 A=A+1
920 IF F3=1 THEN F3=0:LPRINT CHR$(12):GOTO 80
930 LPRINT CHR$(12)
940 IF F6=1 THEN LOCATE0,5:PRINT"Tecle algo para continuar";:T$=
INPUT$(1)
950 IF A=TE THEN CLOSE:GOTO 80 ELSE 750
960 CLS:PRINT"Confirma saída do programa (S/N) ?";
970 C$=INPUT$(1):IF C$="N" OR C$="n" THEN 80 ELSE IF C$="S" OR C
$="s" THEN END ELSE 960
980 FOR O=1 TO C*CO
990 IF EOF(1) THEN CLOSE:GOTO 80
1000 LINE INPUT#1,H$
1010 NEXT O
1020 RETURN
```

Figura 1 - Programa paginador de arquivos-texto no MSX-WORD, sem os recursos de decodificação dos códigos de controle de impressora.

```
10 'PROGRAMA PARA IMPRIMIR
20 'ARQUIVOS-TEXTO DO MSX-WORD
30 'AUTORES:S.G.P.E./P.R.P.E.
40 'DATA: NOVEMBRO DE 1989
50 KEY OFF:LOCATE,,0:CLEAR 1000
60 ON ERROR GOTO 210
70 A$="NÃO":B$="NÃO":A=1:B=8:C=55:F1=0:F2=1:L$="SIM":R=0
80 CLS
90 PRINT"IMPRESSÃO DE ARQUIVOS-TEXTO DO MSX-WORD":LOCATE13,2:PRI
NT"Menu Principal":LOCATE1,4:PRINT"(  ) indicam estado atual das
 funções"
100 LOCATE,6:PRINT"1 - IMPRIME CABEÇALHO (";A$;")"
110 LOCATE,8:PRINT"2 - IMPRIME RODAPE (";B$;")"
120 LOCATE,10:PRINT"3 - NUMERAÇÃO DAS PAGS. (";L$;")
130 LOCATE,11:PRINT"    REDEFINIÇÃO DA NUMERAÇÃO (";R;")"
140 LOCATE,13:PRINT"4 - DEFINE OFFSET DE IMPRESSÃO (";B;")"
150 LOCATE,15:PRINT"5 - DEFINE N' DE LINHAS/PÁGINA (";C;")"
160 LOCATE,17:PRINT"6 - IMPRIME TEXTO"
170 LOCATE,19:PRINT"7 - ABANDONA O PROGRAMA"
180 LOCATE,21:PRINT"Sua opção: ";
190 C$=INPUT$(1):IF C$<"1" OR C$>"7" THEN 80 ELSE PRINT C$:FOR T
=1 TO 100:NEXT T
```

```
200 ON VAL(C$) GOTO 240,300,380,440,510,560,960
210 IF ERR<>53 THEN CLS:PRINT"FOI DETECTADO UM ERRO N':";ERR;"LI
NHA:";ERL:GOTO 230
220 IF ERL=610 THEN PRINT:PRINT"Arquivo não existe":CLOSE:PRINT:
PRINT"Tecle algo":T$=INPUT$(1):RESUME 560
230 CLOSE:PRINT:PRINT"Tecle algo":T$=INPUT$(1):RESUME 80
240 CLS:PRINT"OPÇÃO PARA IMPRESSÃO DE CABEÇALHO"
250 LOCATE,5:PRINT"Confirma impressão de cabeçalho (S/N) ?";:C$=
INPUT$(I)
260 IF CS="N" OR C$="n" THEN A$="NÃO":GOTO 80 ELSE IF C$="S" OR
C$="s" THEN A$="SIM":GOTO 270 ELSE 240
270 LOCATE0,7:PRINT"Digite o cabeçalho (máx.:55 caracteres)"
280 LINE INPUT D$:IF LEN(D$)>55 THEN PRINT:PRINT"Cabeçalho com m
ais de 55 caracteres":PRINT"Tecle algo" ELSE IF D$="" THEN 270 E
LSE 80
290 T$=INPUT$(1):GOTO 240
300 CLS:PRINT"OPÇÃO PARA IMPRESSÃO DE RODAPÉ"
310 LOCATE0,5:PRINT"Confirma impressão de rodapé (S/N) ?";:C$=IN
PUT$(1)
320 IF C$="N" OR C$="n" THEN B$="NÃO":F1=0:L$="NÃO":GOTO 80 ELSE
 IF C$="S" OR C$="s" THEN B$="SIM":GOTO 330 ELSE 300
330 LOCATE0,7:PRINT"Digite o rodapé (máx.:55 caracteres)"
340 LINE INPUT E$:IF LEN(E$)>55 THEN PRINT:PRINT"Rodapé com mais
 de 55 caracteres":PRINT"Tecle algo" ELSE IF E$="" THEN 300 ELSE
 360
350 T$=INPUT$(1):GOTO 300
360 LOCATE0,11:PRINT"Inclúi numeração (S/N) ?";
370 C$=INPUT$(1):IF C$="N" OR C$="n" THEN F1=0:F2=0:L$="NÃO":GOT
O 80 ELSE IF C$="S" OR C$="s" THEN F1=1:F2=0:L$="SIM":GOTO 80 EL
SE 360
380 CLS:PRINT"OPÇÃO PARA NUMERAÇÃO DAS PÁGINAS"
390 LOCATE0,5:PRINT"Confirma nova paginação (S/N) ?";
400 C$=INPUT$(1):IF C$="N" OR C$="n" THEN 80 ELSE IF C$="S" OR C
$="s" THEN 410 ELSE 390
410 LOCATE0,7:PRINT"Digite 0 para não numerar páginas."
420 LOCATE0,9:INPUT"Qual o novo número da página ";A
430 IF A=0 THEN F2=0:L$="NÃO":F5=0:GOTO 80 ELSE IF A>0 THEN F2=1
:L$="SIM":R=A:F5=1:GOTO 80 ELSE 420
440 CLS:PRINT"OPÇÃO PARA OFFSET DE IMPRESSÃO"
450 LOCATE0,5:PRINT"Confirma redefinição de offset (S/N) ?";
460 C$=INPUT$(1):IF C$="N" OR C$="n" THEN 80 ELSE IF C$="S" OR C
$="s" THEN 470 ELSE 450
470 LOCATE0,7:INPUT"Qual o novo offset de impressão ";B
480 IF B>0 AND B<17 THEN 80 ELSE IF B<0 THEN LOCATE0,7:PRINT SPA
CE$(39):GOTO 470
490 LOCATE0,1I:PRINT"CUIDADO: offset alto. Redefine (S/N) ?";
500 C$=INPUT$(1):IF C$="N" OR C$="n" THEN 80 ELSE IF C$="S" OR C
$="s" THEN 470 ELSE LOCATE0,11:PRINT SPACE$(39):GOTO 490
510 CLS:PRINT"OPÇÃO PARA N' DE LINHAS/PÁGINA":LOCATE0,2:PRINT"Má
ximo aceito pelo programa: 55"
520 LOCATE0,5:PRINT"Confirma novo nº de linhas/pág. (S/N) ?";
530 C$=INPUT$(1):IF C$="N" OR C$="n" THEN 80 ELSE IF C$="S" OR C
$="s" THEN 540 ELSE 510
540 LOCATE0,7:INPUT"Qual o nº de linhas/página ";C
550 IF C<0 OR C>55 THEN LOCATE0,11:PRINT"Número de linhas/página
 ilegal":GOTO 540 ELSE 80
```

```
560 CLS:PRINT"OPÇAO DE IMPRESSAO DE ARQUIVOS-TEXTO"
570 LOCATE0,2:PRINT"Indique o drive (A/B) ";:F$=INPUT$(1):PRINT
F$:PRINT:PRINT
580 IF F$="A" OR F$="a" THEN FILES"A:" ELSE IF F$="B" OR F$="b"
THEN FILES "B:" ELSE 570
590 PRINT:PRINT:PRINT"Nome do arquivo:(<RETURN> sai da opção)"
600 LINE INPUT G$:IF G$="" THEN 80 ELSE G$=F$+CHR$(58)+G$
610 OPEN G$ FOR INPUT AS #1
620 CO=1:F4=0:A=1:PRINT:PRINT"Começa impressão na página (1): ";
:
630 INPUT CO:IF CO=0 THEN CO=1:A=1 ELSE IF CO=2 THEN A=2:CO=1:F4
=1 ELSE IF CO>2 THEN A=CO:CO=CO-1 ELSE IF CO<0 THEN PRINT:PRINT"
Valor ilegal":GOTO 620
640 TE=20:PRINT:PRINT"Termina na página (última): ";:
650 INPUT TE:IF TE<CO THEN TE=CO+1 ELSE TE=TE+1
660 IF F5=1 THEN PRINT:PRINT"Numeração redefinida para:";R:PRINT
"Confirma (S/N) ?"; ELSE 700
670 C$=INPUT$(1):IF C$="S" OR C$="s" THEN A=R:GOTO 710 ELSE IF C
$="N" OR C$="n" THEN 680 ELSE 660
680 PRINT:PRINT"Abandona impressão (S/N) ?;"
690 C$=INPUT$(1):IF C$="S" OR C$="s" THEN 80 ELSE IF C$="N" OR C
$="n" THEN 710 ELSE 680
700 PRINT:PRINT"Pausa entre as páginas (S/N) ?";
710 C$=INPUT$(1):IF C$="S" OR C$="s" THEN F6=1 ELSE IF C$="N" OR
 C$="n" THEN F6=0 ELSE 700
720 PRINT:PRINT"Prepare a impressora e tecle algo";:T$=INPUT$(1)
730 CLS:PRINT"{ESC} = sai da impressão":PRINT:PRINT"{STOP} = pau
sa/continua"
740 IF CO>1 OR F4=1 THEN GOSUB 980
750 REM INICIA IMPRESSÃO
760 IF A$="SIM" THEN LPRINT TAB(B);D$ ELSE LPRINT
770 LPRINT
780 X$=INKEY$:IF X$=CHR$(27) THEN CLOSE:GOTO 80
790 FOR I=1 TO C
800 IF EOF(1) THEN CLOSE:F3=1:GOTO 860
810 X$=INKEY$:IF X$=CHR$(27) THEN CLOSE:GOTO 80
820 LINE INPUT#1,H$
830 GOSUB 1030
840 LPRINT TAB(B);K$
850 NEXT I
860 K=I-1
870 FOR L=1 TO 55-K:LPRINT:NEXT L
880 IF B$="SIM" THEN LPRINT TAB(B);E$;"   ";
890 IF F1=1 THEN LPRINT"Pág.:";A:GOTO 910
900 IF F2=1 THEN LPRINT TAB(39);A
910 A=A+1
920 IF F3=1 THEN F3=0:LPRINT CHR$(12):GOTO 80
930 LPRINT CHR$(12)
940 IF F6=1 THEN LOCATE0,5:PRINT"Tecle algo para continuar";:T$=
INPUT$(1)
950 IF A=TE THEN CLOSE:GOTO 80 ELSE 750
960 CLS:PRINT"Confirma saída do programa (S/N) ?";
970 C$=INPUT$(1):IF C$="N" OR C$="n" THEN 80 ELSE IF C$="S" OR C
$="s" THEN END ELSE 960
980 FOR O=1 TO C*CO
990 IF EOF(1) THEN CLOSE:GOTO 80
```

```
1000 LINE INPUT#1,H$
1010 NEXT O
1020 RETURN
1030 I$="":K$=""
1040 FOR X=1 TO LEN(H$)
1050 I$=MID$(H$,X,1)
1060 IF X=1 THEN 1240
1070 IF I$=CHR$(192) THEN I$=CHR$(27)+CHR$(53) Imp.Enfatizada
1080 IF I$=CHR$(193) THEN I$=CHR$(27)+CHR$(52) Desativa Enfatiz.
1090 IF I$=CHR$(194) THEN I$=CHR$(32) FF ==> espaço
1100 IF I$=CHR$(195) THEN I$=CHR$(27)+CHR$(84) Des.Ind./Expoente
1110 IF I$=CHR$(196) THEN I$=CHR$(20) Desat.Expansão
1120 IF I$=CHR$(197) THEN I$=CHR$(14) Expandido
1130 IF I$=CHR$(198) THEN I$=CHR$(18) Desat.Condensação
1140 IF I$=CHR$(199) THEN I$=CHR$(15) Condensado
1150 IF I$=CHR$(200) THEN I$=CHR$(27)+CHR$(45)+CHR$(0) Des.Subl.
1160 IF I$=CHR$(201) THEN I$=CHR$(27)+CHR$(45)+CHR$(1) Sublinhado
1170 IF I$=CHR$(202) THEN I$=CHR$(27)+CHR$(54) Courier Residente
1180 IF I$=CHR$(203) THEN I$=CHR$(27)+CHR$(55) Qual.Carta (RAM)
1190 IF I$=CHR$(204) THEN I$=CHR$(27)+CHR$(72) Desat.Qual.Carta
1200 IF I$=CHR$(205) THEN I$=CHR$(27)+CHR$(71) Qualidade Carta
1210 IF I$=CHR$(206) THEN I$=CHR$(27)+CHR$(83)+CHR$(1) Indice
1220 IF I$=CHR$(207) THEN I$=CHR$(27)+CHR$(83)+CHR$(0) Expoente
1230 K$=K$+I$
1240 NEXT X
1250 RETURN
```

Figura 2 - Programa para paginar arquivos-texto do MSX-WORD incluindo os comandos de impressora (listados na figura, em itálico, não fazendo parte do programa).

```
10 DEFFNP(X)=PEEK(X)+256*PEEK(X+1):KEY OFF
20 CLS:PRINT"Impressão do texto por páginas":TE=INT(FNP(&HC2F1)/
64+.9)
30 LOCATE0,2:LINEINPUT"Numera páginas ? (<CR> = S ) ";A$:IFA$=""
THENF6=1:PA=1ELSEIFA$="N"ORA$="n"THENF6=0:GOTO60ELSE30
40 IFF6=1THENLOCATE0,3:PRINT"Numeração atual é: ";PA:PRINT"Redef
ine (S/N) ? ";:O$=INPUT$(1):PRINTO$:IFO$="S"ORO$="s"THEN50ELSEIF
O$="N"ORO$="n"THEN60ELSE40
50 LOCATE0,5:PRINT"Indique a nova numeração: ";:INPUTPA
60 LOCATE0,7:LP=55:PRINT"Nº de linhas/pág.:";LP:LOCATE0,8:PRINT"
Redefine (S/N) ? ";:O$=INPUT$(1):PRINTO$:IFO$="S"ORO$="s"THEN70E
LSEIFO$="N"ORO$="n"THENLP=55:GOTO80ELSE60
70 LOCATE0,9:PRINT"Nº de linhas/pág. (máx:60): ";:INPUTLP:IFLP>6
0THEN70
80 LOCATE0,11:P1=1:INPUT"Página inicial (<CR>=1)";P1
90 LOCATE0,12:P2=INT(TE/LP)+1:INPUT"Página final (<CR>=última)";
P2:IFP2<P1THEN90ELSEIFP2=0THENP2=1
100 LOCATE0,14:PRINT"Pausa na impressão (S/N) ? ";:O$=INPUT$(1):
PRINTO$:IFO$="S"ORO$="s"THENF7=1ELSEIFO$="N"ORO$="n"THENF7=0ELSE
100
110 LOCATE0,16:CO=1:INPUT"Quantas cópias (<CR>=1)";CO
```

```
120 LOCATE0,18:PRINT"Prepare a impressora e tecle algo.":LOCATE0
,19:PRINT"Se desejar, tecle D para desistir";:GOSUB430:CLS
130 FOR O=1 TO CO
140 IF O>1THEN LP=LP+1
150 IF P1=1 THEN LI=1 ELSE IF P1>1 THEN P1=P1-1:LI=P1*LP+P1:P1=
P1+1
160 V=PA:LP=LP-1
170 LOCATE0,0:PRINT"IMPRIMINDO ARQUIVO TEXTO:"
180 LOCATE0,5:PRINT"Cópia nº";O
190 LOCATE0,15:PRINT"Tecle [ESC] para abandonar impressão":LOCAT
E0,16:PRINT"       [STOP] para pausa/continuação"
200 FOR P=P1TOP2
210 LOCATE15,5:PRINT"Página ";P
220 E$=INKEY$:IF E$=CHR$(27) THEN 420
230 LPRINT
240 LF=LI+LP:U=0
250 FOR I=LI TO LF
260 E$=INKEY$:IF E$=CHR$(27) THEN 420
270 C=64*(INT(I-1)):B=C+FNP(&HD501):X=&HA36A:GOSUB440
280 B=(64*INT(I))-C:X=&HA36D:GOSUB440
290 A=USR5(A)
300 LPRINTCHR$(13):U=U+1:LOCATE0,10,0:PRINT"Linhas: ";U
310 IF I=TE THEN I=LF
320 NEXT I
330 LI=I:LF=0
340 U=U+1
350 FORL=1TO61-U:LPRINT:NEXTL
360 IFF6=1THENLPRINTTAB(39);V
370 V=V+1
380 LPRINTCHR$(12)
390 IFF7=1THENLOCATE0,18:PRINT"Tecle algo para continuar":LOCATE
0,19:PRINT"ou D para desistir":GOSUB430
400 NEXT P
410 NEXT O
420 RUN"MSXWORD3.0"
430 T$=INPUT$(1):IF T$="D"ORT$="d"THENT$="":GOTO420 ELSE RETURN
440 POKEX,B-256*INT(B/256):POKEX+1,INT(B/256):RETURN
```

Figura 3 - Programa "PAGINA.BAS" para ser adicionado ao disco de trabalho do MSX-WORD e rodado junto com o programa.

POR DENTRO DA INTERFACE DE DRIVE

JÚLIO VELLOSO

A principal dificuldade no desenvolvimento de programas profissionais está em relação à gravação e leitura de arquivos.

Tenho visto em diversos profissionais e em quase todos os programas que lidam com drive, (um bom editor de texto ou Page Maker) a impossibilidade de poderem evoluir por causa de um acesso de drive lento ou por não poder acessá-lo de forma direta.

Baseado nesta dificuldade, elaborei um método de acessar o drive com quase o dobro de sua rapidez e de forma a não precisar acessar nenhuma rotina do BDOS, suprimindo esta dificuldade.

Apresentarei neste primeiro artigo sobre interface de drive, a base de conhecimento que julguei necessária para o entendimento dos próximos artigos que em conjunto farão um importante e abrangente documento a respeito deste periférico.

AS ROTINAS BÁSICAS DO DRIVE

Quem já comprou o Livro Vermelho da editora Mac Grow Hill já deve ter notado a existência de duas rotinas que estão mencionadas no livro e que não tem descrição. São elas as rotinas de nome FORMAT e PHYDIO.

A ROTINA PHYDIO

A rotina (PHYDIO) é a mais importante. Com ela se pode fazer qualquer operação em relação o drive (renomeação, gravação, leitura etc)

PHYDIO (144H ou FFA7H): Lê ou grava um ou vários setores no disco que poderá estar em qualquer dos drives (A, B, C e D).

Entrada:

HL: um endereço na memório (0000H à FFFFH)

DE: setor inicial a partir do qual começará a leitura ou gravação:

0 a 719 — 80Tr face simples.
0 a 1439 — 80Tr face dupla.
0 a 359 — 40Tr face simples.
0 a 710 — 40Tr face dupla.

B: número de setores a serem lidos ou gravados.

C: tipo de formatação: 1111 1T0F B

T: 1 — 40 trilhas
0 — 80 trilhas

F: 1 — face dupla
0 — face simples

F (flag C): operação a ser realizada:

0 — leitura de um setor
1 — gravação de um setor

A: drive onde será realizado a operação.

0 — drive A,
1 — drive B
2 — drive C,
3 — drive D

Os endereços seguintes ao especificado por HL deverão conter o setor que será gravado caso a operação que se deseja efetuar seja uma operação de gravação (flag C setada). O buffer apontado por HL deverá ter o comprimento de 512 bytes (que é o tamanho de um setor).

Saida:

F (flag C): indica se ocorreu erro na operação:

setado — sucesso.
resetado — fracosso.

A: indica o tipo do erro ocorrido:

zero — disco protegido
não zero — Erro de E/S.

A ROTINA FORMAT

Ponto de entrada — 147h ou FFACH; depois de chamada a rotina será apresentado para você um quadro de perguntas que deverão ser respondidas, para que a rotina saiba que formatação será efetuada.

Você poderá chamar esta rotina em um ponto interno dela, caso você saiba qual é este ponto, ou tenha uma interface do tipo Microsol.

Esta chamada possibilitará a você pré estabelecer o tipo de formatação, sem ter que entrar no quadro de perguntas.

Para isto, você deverá entrar com a seguinte rotina em linguagem de máquina:

```
        LD HL, 8088H
        LD DE, 51H
        LD BC, 6015H
        LD A, (0FFA8H)
        LD (SLOT), A
        LD A, tipfor
        RST 30H
SLOT    :NOP
PNTINT  :DEFW 401CH
        RET NC
        DEC A
        SLA A
        LD E, A
        LD D,0
        LD HL, TABERR
        ADD HL,DE
        LD A, (HL)
        INC HL
        LD L, (HL)
        LD H,A
        EX DE,HL
        LD C,9
        CALL 5
        RET
```

TABERR:
DEFW ERRM1, ERRM2, ERRM3, ERRM3
DEFW ERRM3, ERRM3, ERRM4, ERRM5
DEFW ERRM3

ERRM1 :DEFB "Disco protegido$"
ERRM2 :DEFB "Não pronto$"
ERRM3 :DEFB "Erro de disco$"
ERRM4 :DEFB "Parâmetro inválido$"
ERRM5 :DEFB "Memória insuficiente$"

Se tudo tiver correto, depois da chamada desta rotina a interface comecará a formatação. Caso contrário, ela emitirá uma mensagem de erro.

Na rotina:

A rotina de impressão usada é a do BDOS. A sua função está descrita na tabela em anexo.
tipfor — conterá o tipo de formatação desejada:

1 face simples 40 trilhas
2 face dupla 40 trilhas
3 face simples 80 trilhas
4 — face dupla 80 trilhas

PNTINT — Neste local terá que existir o ponto de entrada interna da rotina. No caso das interfaces padrão Microsol, o ponto já está pré determinado.

CONFIGURAÇÃO INTERNA DOS DRIVES

Para que se possa mexer com a rotina PHYDIO é bom entender como está dividido o disco em todos os tipos de drive.

A seguir darei uma descrição destas partes, para todos os modelos de drives conhecidos:

FORMATAÇÃO: F8 — 3 1/2 Face simples
SETOR: 0 a 719 ou 0 a 2CEH
CLUSTER: 0 a 354 ou 0 a 167H
1 CLUSTER: 2 setores
FAT #1: 1º setor
FAT #2: 3º setor
DIRETÓRIO: 5º setor
ÁREA LIVRE: 12 ao 719
Nº DE TRILHAS: 80
Tamanho do FAT: 2 setores
Tamanho do diretório: 7 setores
Nº de programas por diretório: 112

FORMATAÇÃO: F9 3 1 2 Face dupla
SETOR: 0 a 1439 ou 0 a 49CH
CLUSTER: 0 a 718 ou 0 a 2CEH
1 CLUSTER: 2 setores
FAT #1: 1º setor
FAT #2: 4º setor
DIRETÓRIO: 7º setor
ÁREA LIVRE: 14 ao 1439
Nº DE TRILHAS: 80
Tamanho do FAT: 3 setores
Tamanho do diretório: 7 setores
Nº de programas por diretório: 112

FORMATAÇÃO: FC — 5 1/4 — Face simples
SETOR: 0 a 354 ou 0 á 167H
CLUSTER: 0 a 354 ou 0 a 167H
1 CLUSTER: 1 setores
FAT #1: 1º setor
FAT #2: 3º setor
DIRETÓRIO: 5º setor
ÁREA LIVRE: 9 ao 354
Nº DE TRILHAS: 40
Tamanho do FAT: 2 setores
Tamanho do diretório: 4 setores
Nº de programas por diretório: 64

FORMATAÇÃO: FD 5 1/4 — Face dupla
SETOR: 0 a 719 ou 0 a 2CEH
CLUSTER: 0 a 354 ou 0 a 167H
1 CLUSTER: 2 setores
FAT: #1: 1º setor
FAT #2: 3º setor
DIRETÓRIO: 5º setor
ÁREA LIVRE: 12 ao 719
Nº DE TRILHAS: 40
Tamanho do FAT: 2 setores
Tamanho do diretório; 7 setores
Nº de programas por diretório: 112

DESCRIÇÃO DAS PARTES DE UM DISCO

BOOT DO SISTEMA: Responsável pela carga inicial do DOS.

FAT 1: Local onde se encontra uma tabela com a relação das partes (clusters), que estão sendo usados pelos arquivos do disco.

FAT 2: Cópia da FAT 1.

DIRETÓRIO DE NOMES: Local onde se encontram: nomes, extensões, tamanhos, clusters iniciais da FAT, datas e horas de criação, que juntos formarão dados importantes para os arquivos.

ÁREA DE TRABALHO: Local onde estão os arquivos guardados no disco.

ÁREA LIVRE: Local livre para os arquivos a serem criados.

AS FUNÇÕES DO BDOS

O usuário já deve estar pensando como o DOS acessa um arquivo, já que só as rotinas anteriores não possibilitam a identificação de que clusters se deverá carregar, ou para onde se destinará os que serão gravados, para compor ou gerar um arquivo.

Para fazer este tipo de operação, o usuário deverá chamar o BDOS (Basic Disck Operation System), que são um conjunto de rotinas. Para acessá las, deverá se ter no registrador C um número, que conterá a identificação da rotina antes da chamada do endereço de número 5:

LD C, número da rotina
CALL 5

O usuário deverá, baseado nos valores de entrada e saída, montar um programa em assembler, juntando as diversas rotinas exibidas na tabela.

Duas rotinas descritas na tabela, pelo simples motivo dos periféricos não existirem, se tornam nulos. São os casos das rotinas de número 3 e 4 para a leitora e perfuradora de cartões.

Nas demais rotinas, existe uma legenda em anexo, para os diversos tipos de dados, que deverão ser entrados nos respectivos registradores, para que fique legível ao usuário, como usar as rotinas descritas

Na tabela em anexo existe uma menção ao endereço do DMA (Disk Memory Acess). Esta refere se ao local da memória onde algumas rotinas do BDOS pegarão ou colocarão as informações do ou para o processamento.

Em todas as rotinas do BDOS, exceto aquelas que lidam com o registro randômico, o FCB deverá ser preenchido até a posição 11. Nas outras posições da FCB, os conteúdos deverão ser preenxidos com zero:

```
FCB: DB 0
     DB "NOMEPGM EXT"
     DB 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
     DB 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
     DB 0, 0, 0, 0, 0
```

REFERENTE A TABELA

As rotinas marcadas com um asterisco (*) não estão presentes na última versão do cartucho da Microsol.

A rotina de escrita randômica de bloco se comporta do mesmo modo que a 27, exceto pela operação de escrita e indicação de disco protegido.

A rotina de escrita randômicas com zeros funciona do mesmo modo que a rotina 22h, exceto que, sempre quando uma escrita elimina um arquivo de for

hex		Entrada	Saida
0	Inicializa a CP/M		
1	Lê um caracter da consale (CON)		A = CODASC
2	Envia um caracter aa cansole (CON)	E = CODASC	
3	Lê um caracter da leitara (RDR)		A = CODASC
4	Envia um caracter à perfuradora (PUN)	E = CODASC	
5	Envia um caracter à impressara (LST)	E = CODASC	
6	Entrada direta na consale	E = FFH	A = BYTE
	Saida direta na cansale	E = CODASC	
7	Lê a IOBYTE		A = BYTE
8	Preenche a IOBYTE	E = BYTE	
9	Imprime string	DE = STRING	
0A	Lê a buffer da console	DE = BUFFER	no buffer
0B	Verifica a estado da consale		A = ERRO 1
0C	Farnece a número da versão em usa		HL = versão
0D	Inicializa as unidades de disquetes		
0E	Seleciona drive para usa	E = NDRV	
0F	Abre arquiva	DE = FCB	ERRO2
10	Fecha arquiva (atualiza diretória)	DE = FCB	ERRO2
11	Procura primeira acorrência	DE = FCB	ERRO2
12	Pracura próxima acarrência	DE = FCB	ERRO2
13	Elimina arquiva	DE = FCB	ERRO2
14	Leitura seqüencial de um registro	DE = FCB	ERRO2
15	Escrita seqüencial de um registra	DE = FCB	ERRO3
16	Criação de um arquiva	DE = FCB	ERRO3
17	Altera a name de um arquiva	DE = FCB	ERRO3
18	Indica as drives que estão ativas		HL = LIGDRV
19	Farnece a drive carrente		A = NDRV
1A	Indica endereça da DMA	DE = FILERAM	
1B	Dá o endereço do mapa de alocação		HL = END
1C	Pratece a disquete cantra gravação		
1D	Indica as unidades pretegidas		HL = LIGDRV
1E	Designa atributas (DIR, SYS, R/O, R/W)	DE = FCB	A = ERRO2
1F	Endereça da tabela de parâmetros		HL = END
20	Define a códiga da usuária (0..31)	E = BYTE	
	Obtém o código da usuária (0..31)	E = FFH	A = BYTE
21	Leitura randômica de um registra	DE = FCB	A = ERRO4
22	Escrita randômica de um registra	DE = FCB	A = ERRO4
23	Tamanha da arquiva (em setares)	DE = FCB	na FCB
24	Pasiçãa randâmica da registra	DE = FCB	no FCB
25	Reinicia de drive		
26	Escrita randômica de blaco	DE = FCB HL = cantador	A = ERRO3
27	Leitura randâmica de blaca	DE = FCB HL = contador	A = ERRO3
28	Escrita randâmica cam zeros	DE = FCB	A = ERRO4
* 29	Nãa implementada		
2A	Retarna a Data		HL = ana DE = mês/dia A = semana
2B	Modifica a Data	HL = ana DE = mês/dia	
* 2C	Retarna a Hora		HL = hora/min DE = seg/c.seg.
* 2D	Modifica Hora	HL = hora/dia DE = seg/c. seg	
2E	Verificaçãa de escrita		A = ERRO3
2F	Leitura de setares	DE = setor HL = 00/Nº set	A = ERRO3
30	Escrita de setores	DE = setor HL = 00/Nº set	A = ERRO3

ma descantinua, o espaco intermediário é preenchido cam zeros.

A seguir apresento a legenda referenciada anteriarmente:

CODASC— código asc para caracter (varia de 0 a 255)

ERRO1 — 00: inativa
FF: caractere dispanivel

ERRO2— FF: arquivo não encontrado
00...03: atributo do arquivo encontrado (DIR,SYS,R O ou R/W

ERRO3— 00: operacão OK,
00: encontrou fim de arquivo (leitura)
: disca cheia (gravação)
: diretório cheio (criação de arquivo)
: erro de escrita (verificacão de escrita/escrita setores)
: erra de leitura (leitura de setores)

ERRO4— 00: operacão OK
01: tentativa de ler um registro que não foi gravado
02: não usado
03: não pode fechar tensão corrente.
04: tentativa de ler uma extensão não gravada.
05: diretório cheio.
06: leitura após fim do disquete.

FCB— endereço inicial do FCB (File Control Block):

00: drive —00 = corrente, 1 = A, 2 = B,...)
01...08: nome do arquivo
09...11: extencão do arquivo
12: número da extencão
13...14: ————————
15: número de registra nesta extencão.
16...31: relação dos blaca alocados nesta extencão.
32: próximo registro a ser lido ou gravado nesta extenção.
33...34: número do registro randômico.
35: indicador de overflow de registro randômico.

BYTE— valor de 8 bits.
END — valor de 16 bits (endereço).
NDRV — número do drive (0, 1, 2, 3 ou 4).
FILERAM — local onde será ou está o bloco (128 Bytes) a ondo será corregada ou gravado.
LIGDRU — Corresponde a um número binário de 16 posições, em que coda uma corresponde a um drive, nomeados nesta ordem:

P O N M L K J I H G F E D C B A
X X X X X X X X X X X X X X X X

Onde:
X — 0 — desligado.
1 — ligado.

ENTREVISTA

ANTONIO NASSER, engenheiro químico aposentado, ingressou no campo da informática, mais especificamente no ramo dos micros MSX, quase que por acidente.

Começando como usuário que procura e adquire o melhor para seu equipamento, fundou a MSX SOFT do Rio de Janeiro. Evoluindo constantemente, a MSX SOFT, hoje, possui uma das maiores redes de distribuição de hardware para MSX.

Nesta entrevista, Nasser nos dá a chance de avaliar e discutir melhor suas idéias, além de adiantar alguns lançamentos.

Há quanto tempo trabalha com o MSX?

Trabalho com este tipo de equipamento desde o seu surgimento. Aliás, a MSX SOFT existe desde essa data.

Você acha que, hoje em dia, a indústria de hardware já oferece tudo o que o MSX precisa ou ainda tem alguma coisa a mais?

Tem muita coisa mais. Nós mesmos estamos pretendendo lançar algumas novidades em termos de hardware, juntamente com a DDX. Uma dessas novidades que está praticamente na iminência de ser comercializada é um disco Winchester de 2 Mbytes para o MSX. Calculo que em fevereiro, no máximo em março, este equipamento já esteja disponível no mercado.

Independente disto, posso adiantar que ainda existe cerca de meia dúzia de novidades de hardware sendo desenvolvidas para o MSX no Brasil.

Quer dizer que estão tentando profissionalizar ainda mais o MSX?

Sim. Aliás, estamos contando com todo o suporte da Gradiente, que nos fornece qualquer informação que necessitemos sobre o MSX. Apesar de mantermos um relacionamento bem estreito com a Gradiente, os projetos estão sendo desenvolvidos exclusivamente por nós, ou melhor, nós e a DDX.

A MSX SOFT é uma das empresas que fazem a transformação do MSX 1 em MSX 2. Como funciona este esquema?

Nós somos uma empresa intimamente ligada com a DDX. Fora a Gradiente, poderia até afirmar que a DDX é a segunda empresa nacional que mais se dedica ao MSX, no campo de hardware. Como seus representantes, estamos associados e todos esses desenvolvimentos são feitos em conjunto com a DDX.

Com relação à transformação do MSX 1.0 para MSX 2.0, simplesmente fazemos a instalação, pois a placa é desenvolvida pela DDX.

Qual é a reação, a receptividade por por parte do comprador?

Muito boa e não temos tido reclamações no sentido da utilização.

Entretanto, é preciso saber se vale a pena. É lamentável que a utilização do MSX 2 seja unicamente voltada aos jogos. Em termos de aplicativos, de programa profissionais, quase nada existe para o MSX 2. Portanto, não vale a pena fazer a transformação se o uso que pretende se dar à máquina é estritamente profissional. Para isso pode-se utilizar a versão 1, pois a transformação por si só não profissionaliza mais o computador. Para o profissional basta a versão 1.

Em termos de hardware, pelo que podemos observar, a MSX SOFT pode ser considerada uma firma sui generis?

Sim, porque nós e a DDX, inclusive posso falar pelas duas, somos in-

teiramente voltadas para a linha MSX. Nos dedicamos exclusivamente ao MSX. Nada que a DDX fabrique esta fora do campo do MSX. Somos especializados. A DDX na Indústria e nós na comercialização.

A MSX SOFT atua, também, no mercado de compra e venda de micros usados?

Certa vez, uma empresa pretendia usar micros compatíveis com o MSX como terminais de PC. Entrou em contato conosco para que pudessemos adquirir os equipamentos usados para tal fim e para avaliar, também, o estado de cada um, fazendo os reparos que talvez fossem necessários.

Anunciamos que estávamos comprando micros usados e, quando já tinhamos cerca de dez computadores comprados, a politica de tal empresa mudou, o que nos obrigou a ficar com as máquinas.

Com isso, tivemos que vender os aparelhos usados, como usados. Foi um bom negócio mas, lamentávelmente, tivemos que parar, pois não é nossa meta trabalhar com usados.

Que incentivo a MSX SOFT oferece em relação ao software?

Nós não temos pessoal voltado somente ao desenvolvimento propriamente dito. Trabalhamos com software porque o mercado exige, mas não temos intenção em desenvolvê-lo especialmente para empresas e afins, mesmo porque nos falta material humano.

Só gostamos de fazer as coisas bem feitas e é muito difícil encontrar pessoas que tenham um conhecimento profundo do assunto. Quando você desenvolve um programa para uma empresa, além do trabalho para criá-lo, existe o suporte e a manutenção que, em muitos casos, pode ser interminável. Para isso, é preciso pessoal especializado que, como disse, não temos.

O que fazemos nesta área é trabalhar com joguinhos e programas já prontos, como o SuperCalc e o dBase II.

Temos um bom trabalho em software, mas a nossa especialidade, hoje, é hardware.

Caso algum programador procure a MSX SOFT, o que irá encontrar?

Seria bem recebido, mas teria que fornecer o suporte necessário ao usuário.

De vez em quando aparecem alguns autores, mas que querem apenas comercializar o produto, vender e até logo, o que não aceitamos, porque sempre surgem dúvidas cujo o esclarecimento depende, exclusivamente, do autor.

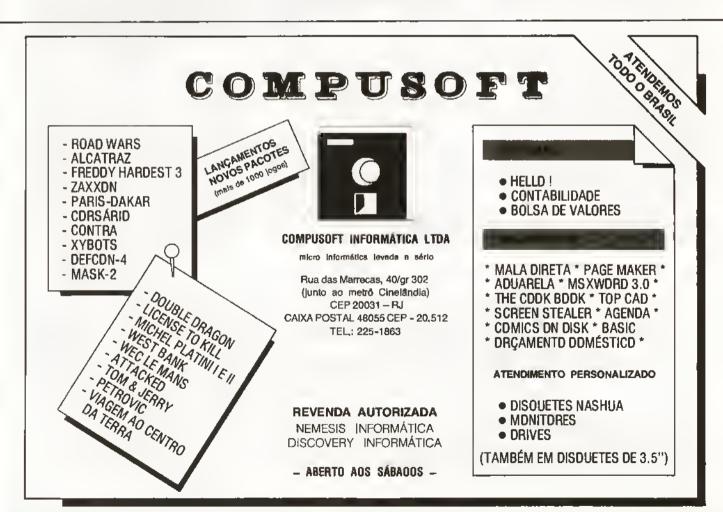
Existe uma grande distância entre o manual e o programa. O manual é insuficiente. Até em programas bem documentados, como o dBase II, há casos de dúvidas e pontos obscuros, não cobertos pelo manual.

Já em termos de hardware, temos condições de dar todo o suporte, pois temos todo o conhecimento necessário para este fim.

Um assunto muito polêmico é a pirataria. Como vai a pirataria aqui e lá fora?

Bem, eu não conheço pirataria. Conheço pessoas que trocam jogos, pessoas que vendem jogos e pergunto se isso é pirataria.

Também conheço pessoas que recebem jogos, tiram o nome do autor e colocam seu próprio nome.

Não sei quem você classificaria de pirata. Se você considera pirata

aquele que vende programas e aquele que coloca o próprio nome, então todos nós somos piratas, pois quem não faz uma coisa, faz outra.

Então, quem revende software copiado é pirata?

Sim, porque tais programas nunca são nacionais. Ou são japoneses ou ingleses. Aqueles que revendem programas e dizem que são nacionais é porque trocaram o nome do verdadeiro autor. Se encararmos por este ângulo, podemos nos considerar todos piratas

E quanto aos verdadeiros autores nacionais? Não seriam estes atingidos mais diretamente, além de não terem como recorrer?

Sejam os autores de onde forem, brasileiros ou não, há direitos envolvidos.

Mas, neste momento, pergunto quem são os autores brasileiros? Você pode apontar autores brasileiros autênticos? Mesmo que existam, eles estão de mãos amarradas, como você mesmo disse.

Na minha opinião, o que não se pode fazer nada para corrigir, está certo. Se está errado, pode-se fazer algo para consertar. Esta é a minha filosofia de trabalho.

Pirataria ocorre em todas as áreas, seja na informática, no cinema ou na música. Infelizmente, esse negócio de direitos não funciona no Brasil.

Você fala de pirataria de software, que é mais evidente, mas a pirataria de hardware é muito mais grave. Todos se preocupam com a pirataria de software, criticam, rotulam e ninguém está atento à pirataria de hardware, que é muito mais danosa e muito mais prejudicial.

O desenvolvimento de um equipamento envolve muita coisa. Uma indústria que se dispõe a criar algo, gasta tempo e dinheiro em técnicos, testes, criação, adaptação e tudo mais que envolve o lançamento de um artigo novo. Quando, finalmente, o produto está no mercado, aparece outra empresa, que só observou, compra o produto na loja, abre e copia. Copia literalmente, todos os detalhes, desde a embalagem até o manual. Aí está a pirataria e ninguém a chama de pirataria. Ninguém critica e é apenas mais um concorrente no mercado. Ninguém faz menção aos piratas de hardware. Já li em certas revistas sobre o pirata de software, mas nada a respeito de piratas de hardwere.

Isto não poderia ser explicado por falta de divulgação?

Todo mundo sabe que existe mas, talvez por ser uma indústria, as revistas tenham medo de mexer com eles. Tudo bem quando se trata de software, mas, quando é uma indústria, que pode ter uma organização jurídica, não há manifestação. Nunca ouvi dizer que alguém tenha acionado esta ou aquela empresa por pirataria. Por isso que ainda digo que não sei quem é pirata.

Qual é o seu ponto de vista do mercado atual?

Sinto que o mercado está bom, aquecido. Inclusive, ouvi dizer que um novo MSX está para ser lançado em breve por uma firma paulista. Tomara que seja verdade, mesmo porque parece que eles estão optando pela fabricação direta do MSX 2. Só lamento que a Gradiente - não tenha produção suficiente para atender a demanda do mercado.

Se alguém quiser adquirir um computador para colocar em casa ou no escritório, não há outra opção senão o MSX. Antigamente, existiam o CP 400 e o CP 500 da Prológica, que não são mais fabricados. O Apple morreu. Então, além do PC, cujo o investimento é proibitivo, resta o MSX.

O lançamento deste novo MSX já tem data?

Só sei que um novo MSX, compatível com a versão 2, está preparado e é só.

O novo governo promete fazer uma mudança na área de Informática, o que pode amenizar a reserva ou mesmo fazê-la cair. Qual é a sua opinião neste assunto?

Acredito que a reserva de informática possa até cair pois, como dito pelo nosso presidente, parece que as portas do mercado serão abertas para a informática. Eu não entendo muito de política, mas acredito que, quando você não sabe de alguma coisa, o melhor caminho é procurar alguém que saiba, até que você aprenda e não precise mais desta pessoa. Acho que, se a reserva terminar, nós vamos aprender muito com os americanos, ingleses e japoneses e, depois, poderemos produzir sozinhos, a exemplo do que aconteceu com o mercado automobilístico há algumas décadas atrás.

Realmente, hoje em dia, a indústria automobilística nacional é bem significativa mas, na época desta abertura, várias fábricas nacionais que estavam surgindo não agüentaram a concorrência e acabaram fechando?

Sim, mas eram fábricas que produziam carros não confiáveis, de baixa qualidade. Além disso, a tecnologia desenvolvida aqui é muito cara. Quando há a possibilidade de dispor de tecnologia de outro país, fica muito mais barato. Hoje, o Brasil produz carros com preços competitivos até no exterior, o que possibilita a sua exportação. Os computadores nacionais custam, uma fortuna, comparados com os computadores estrangeiros, além de serem de baixa qualidade, porque nós não sabemos fazer.

CPM

Se você se interessa por programação profissional, entre para o CPM, que vem a ser um clube voltado apenas para a troca de programas, utilitários, aplicativos e educacionais. Se você montou um programa do tipo citado acima, entre também em contato conosco.

CPM — CLUBE PADRÃO MSX
A/C RONILDO
CAIXA POSTAL 370 —
AGÊNCIA GUANABARA
67000 — ANANINDEUS — PA

POKES E PEEKS

Antes de tudo, parabenizo a CPU pela qualidade da revista, mas como ninguém é perfeito, tenho algumas críticas.

Os usuários menos privilegiados ficam boiando em alguns artigos. Acho que os programas editados na revista deveriam ter informações mais técnicas. Outro problema para o leitor é quando falam de pokes e peeks. Essa matéria deveria ser mais comentada, dando mais explicações sobre sua utilização em programas. Se não fosse incômodo, gostaria que me mandassem dados sobre a linguagem Cobol do MSX.

MARCELO CASTANHA
R. JACARAPINHA, 345
02415 — SÃO PAULO — SP

Procuramos sempre passar o maior número de informações aos nossos leitores. Contudo, certos artigos são publicados para leitores com um nível de conhecimento mais alto, sendo que sempre procuramos atender a toda a faixa de leitores.

A linguagem Cobol será comentada em um dos próximos números de CPU.

PROGRAMAS

Escrevo-lhes, primeiro, para prabenizá-los pelo alto nível instrutivo que a revista vem mantendo.

Possuo um programa que elaborei em conjunto com um amigo e gostaria de comercializá-lo. Há alguma softhouse interessada em promover a criação de programas?

Tenho certas idéias que talvez fossem interessantes para os leitores de CPU. Gostaria de saber se há remuneração e, em caso afirmativo, qual é a média.

Gostaria de me comunicar com outros possuidores de MSX.

SÉRGIO CARDOSO SANTOS
RUA SENADOR VERGUEIRO, 193/603
FLAMENGO
22230 — RIO DE JANEIRO — RJ

A melhor forma para comercializar seu software é entrar em contato com as várias softhouses que comercializam programas de autores nacionais e ver qual delas lhe oferece melhores condições.

Os artigos publicados em CPU são remunerados mas não há uma média. Envie-nos sua colaboração para análise, que entraremos em contato, dando todas as informações.

ANALISANDO

Achei interessante a idéia de consultarem o leitor. Devo reconhecer que a revista CPU é uma das melhores nacionais para MSX. Porém, há muito para melhorar. Uma revista como a MSX Magazine ou a MSX Club traz muito mais informações sobre novidades que a CPU. Muitos pensam que isso é lógico, pois as coisas sempre aparecem por lá antes de aparecerem por aqui. Mas aqui também pintam novidades, como a MEGARAM e a transformação para MSX 2, projetos nacionais. Outra vantagem nas revistas importadas é que, geralmente, encontramos uma seção sobre eletrônica, onde existem pequenas coisas que um usuário pode montar em sua casa ou serviço. Coisas que vão desde um Digitalizador de Imagens (Magazine 12/88) até uma interface MIDI para o MSX (Magazine 10/89). A revista CPU pode coletar e selecionar coisas do gênero para publicar.

Um ponto negativo foi a entrevista com Renato S. Oliveira, pois não há como concordar com certos pontos que levam o usuário a pensar de maneira errada.

O AX não passa de um PC Japonês incrementado. Os PCS japoneses concorrem com o MSX há muito tempo e não será gora a vitória de algum. Foi um erro da Folha

de São Paulo publicar a matéria sobre o AX da maneira como publicou. Talvez tenha sido por isso que dias depois foi publicado o aumento do Sistema MSX no mundo,. Se alguém quer micro para derrubar o MSX, este deve procurare algo como o X68000, um computador com 16 Mbytes de RAM, altíssima resoluçaão gráfica, mas de 30 Mhz, 65535 cores simultâneas, 32 canais de som estéreo. Sua única desvantagem é o preço.

Não se pode dizer que um computador é melhor que outro sem analisar todas as vantagens e todas as desvantagens, como também o que precisamos fazer. Caso contrário, poderemos comprar um equipamento caro que não faz a metade daquilo que necessitamos. Um exemplo disso, para quem precisa de um equipamento para trabalhar na área gráfica, é o MSX 2+, que muitas vezes é deixado em segundo plano e para muitos vale mais que uma Amiga.

Outro ponto da entrevista que me intrigou foi quando o Sr. Renato disse que os novos Expert Plus virão com os CIs soldados na placa e afirmou que seria bom para o usuário. Agora, pergunto: Como faremos para colocar uma placa Turbo ou uma placa de transformação para MSX2 e outros periféricos internos?

RAFAEL ARAUJO DA SILVA
RUA DA BANDEIRA, 131
BLOCO B APTº 13
CABARAL
82500 — CURITIBA — PR

DICAS DO LEITOR

Sou assinante da revista CPU e gostaria de lhes parabenizar pelo excelente trabalho que estão fazendo. Conheço outras publicações para o MSX, mas nenhuma que leve tão a sério os temas apresentados, tanto jogos como linguagens e aplicações profissionais, e auxilie o usuário como a CPU tem feito. Fica de parabéns também a equipe que transmite programas de linguagens basic, pascal, assembler, etc. visando o uso mais profissional e sério pelo usuário. A única queixa feita é quanto ao dBase II que parece abandonado. É uma feramenta de mais fácil aprendizado depois do pascal, para o usuário autodidata, que creio ser a maioria.

Gostaria de propor uma nova sessão que estimularia os usuários a pensar e competir. Esta daria um tipo de troféu ao vencedor, do tipo disco com programas ou um programa escolhido previamente, etc. Os problemas "jogados" aos usuários seriam em nível de programação e resolução de pequenas funções no MSX, como: programar as teclas de funções em ambiente dBase II; programar as mesmas teclas e as setas cursoras em Pascal para serem utilizadas com o Word Star, Supercalc; ler e mudar o volume do diskete direto do sistema operacional, por autoexec. bat, etc.

Outra proposta seria a criação de cursos do tipo "como fazer", como cursos de assembler (como digitar, gravar e ler os programas editados em revistas), cursos em Pascal (como fazer agendas, cartas, etc), cursos em dBase II, etc.

Há algumas dúvidas que eu gostaria que vocês me ajudassem. Gostaria de obter a listagem de um programa em Pascal publicado no número 4, sobre Inteligência artificial (a minha listagem veio ao contrário e invertida, e não consegui ninguém que me forneça a mesma) Queria, se possível, poder adquirir os números 1, 2, 3, 5 e 6, ou xerox destes. Vocês poderiam publicar dicas e mapa do Avenger (Samurai) e do Survivor? Qual o poke para fazer voltar o click do teclado?

Há alguém que ajude a programar a Megaram 128k da Orionsoft para ser usada como ramdisk com programas de arquivo além de vários programas em basic?

Envio algumas dicas sobre o cursor no dBase II e para os usuários que fazem uso da Praxis 2 como impressora.

Cursor em dBase II Plus — MSX

Para se conseguir um cursor piscante no dBase II Plus, devemos digitar POKE 64682,1.

Para desligar o piscar do cursor, deve-se fazer: POKE 64682,0. O cursor continuará a piscar mesmo ao sair do dBase, e entrar no SC 2 ou Word Star 3.0.

Ainda no dBase II, se digitarmos POKE 64683,0 teremos as letras sempre maiúsculas. POKE 64683,1 desativa.

Acentuação do MSX Word 3.0 na Praxis 2

O usuário deverá listar o programa MSXWORD3.0 e chamar a linha de número 912, onde deverá ser digitado a ordem para ativação da tabela Ablcomp (no caso da ABNT — tabela extra), como se segue:

— 912 FOR I=1TOF:CLS:LPRINT CHR$(27)+"Z":PRINT:PRINT:etc (o resto da linha não é alterada).

Assim, sempre que a opção "p" for utilizada, a tabela da Praxis será acionada e não haverá erro na acentuação.

Peço aos usuários da Praxis que divulguem dicas do seu uso, como por exemplo: como fazer negrito na Praxis, etc.

Curiosidades sobre Abadia Del Crímen

Na tela do monitor surge, no lado inferior esquerdo, palavras que nos

mostram o tempo disponivel e a relação com os principais eventos. No livro, há o relato de que o manuscrito de Adso (o garoto), é dividido em 7 dias. Creio que seja o tempo máximo para resolvermos a charada.

- matinas — hora entre 2h30min — 3:00h da madrugada
- Laudes — 5 e 6 horas da manhã
- Primeira — 7:30h, o raiar do sol
- Terceira — 9:00h
- Sexta — meio-dia, hora da refeição
- Nona — 14:00 a 15:00h
- Vésperas — 16:30h, ao por do sol hora da ceia
- Completas — 18:00 às 19:00h, hora de se recolher.
- noche — 19:00 às 02:00h

O frei Guilherme, quando na abadia, deve sempre respeitar o sono, com exceção do dia 05, quando deverá arriscar entrar com a lamparina e Adso na biblioteca.

Acentuação em DBase II para usuários da Praxis 2

Pode-se utilizar o programa Filtro, que vem junto no diskete do dBase II Plus, em arquivo batch. Para os que não querem gastar espaço do disco ou não possuem a nova versão do dBase, pode-se fazer:

— Usar o modify comand e ir ao ponto onde é escolhido a forma de apresentação (impressão ou video; após o "case" ou "if" impressão ON, por:

```
SET PRINT ON
? CHR$ (27) CHR$ (9)
SET PRINT OFF
```

Em um dos próximos números de CPU iremos publicar um novo programa de linguagem artificial, mais aprimorado do que o anterior.

Os números de 1 a 8 de CPU estão esgotados e, devido à situação econômica atual, não temos previsão de quando serão reimpressos.

O mapa do jogo Avenger já está sendo elaborado e em breve será publicado.

TROCA DE CORRESPONDÊNCIA

Gostaria de me corresponder com outros leitores desta revista, que queiram trocar jogos, aplicativos, utilitários, dicas e manuais para os mesmos. Possuo um MSX 1.1, drive e 5 1/4", gravador, Megaram monitor, impressora e uma grande quantidade de programas.

RICARDO BORGES GOMIDE
CSA 03 LOTE 16 APTº 602
72015 — TAGUATINGA — DF

• • •

Quem quiser trocar dicas, programas e bater um papo por carta, escreva-me.

FABIO VILLAVICENCIO
CAIXA POSTAL 717
95700 — BENTO GONÇALVES — RS

• • •

Estou escrevendo para parabenizá-lo pela excelente matéria publicada em CPU número 10 entitulada APPLE VERSUS MSX, pois foi interessante e totalmente verdadeira. Gostaria, também, de trocar informações e tudo mais com os outros leitores da revista e usuários de MSX.

CARLOS HENRIQUE SUFFERT
R. RICARDO GAVENKI, 55
91000 — PORTO ALEGRE — RS

• • •

Venho, através desta, parabenizá-los pela revista CPU, que acompanho desde o número dez, sendo que gostaria de comprar os números de 1 a 9. Também gostaria de trocar dicas e maceles de jogos como Freddy Hardest II, Rambo I, Mask III, Alcatraz, The Castle Excelent, Pay Load, etc. Interessados, telefonem para (061) 441-2698 nos fins de semana.

TACITO AUGUSTO R. DE FREITAS
R. CEL. JOSÉ DA COSTA, 27
76860 — ORIZONA — GO

• • •

Desejo trocar programas, jogos, etc., com outros leitores. Possuo um MSX e disco.

LÚCIO SOARES DA SILVA
QI 14 — CONJUNTO D, CASA 15 — GUARÁ I
71015 — BRASILIA — DF

• • •

Possuo um micro Expert 1.1 drive DDX 5 1/4", impressora e monitor e gostaria de entrar em contato com usuários da linha MSX, para a troca de programas, dicas, livros, revistas e pokes. Tenho uma coleção de mais de 750 jogos e 50 aplicativos.

HDI — SOFT CLUB MSX
AV. TAMANDARÉ 654 —
ED. CANNES APTº 101 —
TAMBAÚ — 58033 —
JOÃO PESSOA — PB.

... e falando em número anterior...

TESTE "t" DE STUDENT PELO COMPUTADOR

APLICAÇÕES DE USO GERAL

Corrigindo alguns erros constantes do artigo acima, publicado na revista CPU II (nº 14): 24 — 32, 1990.

Página	Coluna	Linha	Erro publicado	Correção a fazer
24	02	15	"menor que ..."	"maior que..."
25	01	18	"não significante" (N. S.)	"Não Significante" (N. S.)
25	01	28	(p 0.05)	($p < 0.05$)
25	01	48	a-se p 0.05	a-se $p < 0.05$
25	01	50	b-se p 0.05	a-se $p > 0.05$
26	02	Título	O TESTE "t" EMPAREL PAREADO (paired "t" test) HADO OU	O TESTE "t" EMPARELHADO OU PAREADO (paired "t" test)

MSXDEBUG E SCREEN IV

Os programas que faltavam na sua biblioteca

MSXDEBUG

1) **DISP** endereço
Mostra o conteúdo da memória dado por endereço. Para avançar ou retroceder, utilize as teclas dos cursores. Estando na posição desejada, pressione <CR>. Neste momento, você poderá alterar o conteúdo desta memória ou se deslocar ao longo dos códigos livremente. Para alterar utilizando diretamente o teclado, pressione <CR> novamente. Repita sempre que precisar mudar o modo de edição. Para sair, basta pressionar <ESC>.

2) **EXEC** endereço
Executa este endereço sem break-points.

3) **MOVE** início, fim, destino
Move um bloco na memória. Inicio deve ser menor que fim.

4) **FILL** início, fim, dado
Preenche uma área de memória especificado por inicio e fim.

5) **DIR**
Mostra o diretório do disco de forma simples.

6) **DSAVE** nome. ext, início, fim
Salva o bloco limitado por início e fim no disco com o nome nome.ext. Salva qualquer coisa em qualquer lugar.

7) **DLOAD** nome. ext, endereço
Carrega qualquer arquivo, desde que esteja no disco e o coloca no endereço dado por endereço. Nestes três comandos não é possível especificar o drive. Será utilizado aquele que estava habilitado antes de executar o MSXDEBUG.

8) **DOS**
Encerra as atividades no MSXDEBUG e retorna ao DOS.

SCREEN IV

Comando	Descrição
SCREEN 4	: Habilita tela de 64 x 24 linhas
FILL	: Um paint 8 vezes mais rápido
OVER	: Sobrepõe textos
CLS ON/OFF	: Inibe e habilita o CLS quando se muda o número de colunas com o comando WIDTH
WIDTH	: Alterado para comportar 64 colunas
SCREEN SAVE/LOAD	: Grava e carrega telas do cassete
SCREEN OUT	: Imprime telas na impressora
LIST	: Apresenta a listagem de um programa, sendo que a mesma é interrompida quando a tela fica cheia, pedindo que se pressione uma tecla pra prosseguir.

Solicite logo seu disco com os dois programas. O preço é de 21 BTNF's do dia de postagem do pedido.

Seu software tem garantia de 30 dias contra qualquer defeito de gravação.

Acompanha o programa extenso manual.

Envie seus dados para a revista CPU, juntamento com cheque nominal à Águia Informática Ltda. ou vale postal pagável na agência Copacabana.

A entrega é imediata e as despesas de remessa, através de correspondência registrada, já estão incluídas.

REVENDEDORES:

MSX TRONIC INFORMÁTICA LTDA.
Rua Senador Vergueiro 207/1205 - Flamengo - 22230 - Rio de Janeiro - RJ
Tel.: (021) 552.4581

GAME OFF TIME SOFTWARE E HARDWARD
Av. Jabaquara 1598 sala 08 - Saude - 04046 - São Paulo - SP
Tel.: (011) 581.2739

MANÍACOS DO MSX
Av. Jabaquara 99 sala 54/5º andar - Vila Mariana - 04050 - São Paulo - SP
Tel.: (011) 276.7665

constante entre o DOS e o próprio MSXDEBUG. A área de memória a partir de 0D000H, mais precisamente por volta de 0D600H, está ocupada justamente com as rotinas de suporte do DOS ao MSXDEBUG. Por isso, preenchendo aquela área com valores aleatórios, o sistema perde o controle já que as entradas das rotinas do DOS foram destruídas.

Caso você precise de mais espaço para trabalhar, use apenas 1 drive e mantenha pressionada a tecla CONTROL ao ligar o computador. Com isso haverá um ganho de cerca de 1K.

DRAW E PUT SPRITE

Sou um leitor assíduo da revista CPU. Nós e a linha MSX estávamos precisando de uma revista de nível profissional, com uma linguagem moderada.

Gostaria de dar algumas sugestões. No final de cinco ou dez revistas publicadas, editar uma edição extra com as respostas das perguntas mais interessantes e as mais frequentes. Deveria falar, também, sobre periféricos, jogos mais comprados e sobre a participação da linha MSX no mundo.

Gostaria de saber como posso fazer para que uma figura geométrica qualquer apareça por inteiro e instantaneamente, como acontece nos jogos e sem usar o PUT SPRITE, pois o número da chamada só é permitido de 0 a 31. Usando o DRAW, por exemplo, os pontinhos aparecem progressivamente.

Na revista número 1 apareceu uma informação errada na página 28, dizendo a senha dos Goonies incorretamente.

FABIO DIAS GOLDSTEIN
RUA PRES. CARLOS DE CAMPOS, 183/104
LARANJEIRAS
2231 — RIO DE JANEIRO — RJ

Os únicos recursos que o BASIC oferece para colocar uma figura na tela são, justamente, o DRAW e o PUT SPRITE. Infelizmente, devido à lentidão desta linguagem, não é possível criar uma impressão instantânea na tela. As figuras que "aparecem" instantaneamente como acontece nos jogos, foram feitas por rotinas em linguagem de máquina. Em ambos os casos, a impressão é feita por partes, mas a diferença está na velocidade, que é bem maior nas rotinas feitas em linguagem de máquina.

A criação de rotinas para impressão de figuras utilizáveis pelo Basic já está sendo estudada pela equipe da revista. É só aguardar.

Existem várias versões do jogo Goonies e cada uma delas possui senhas diferentes.

ENUMERANDO

Quero, primeiramente, parabenizá-los pela excelente revista que é a CPU.

Tenho algumas dúvidas, sugestões e dicas em relação ao MSX e à revista. Primeiro, as dúvidas:

1 — Eu entendo que um aplicativo ou um utilitário não rodem em fitas cassete, mas qual a razão de jogos como Nemessis I (sem Megaram, Elite, Operation Wolf e outros não rodem em fitas, já que o jogo é carregado apenas uma vez para ser executado?

2 — Os cartuchos para o sistema MSX tem uma memória limite? Qual a capacidade?

3 — Um jogo como Nemesis e outros ditos especiais (Abadia Del Crimem, etc., podem ser gravados em cartuchos?

4 — Eu poderia ter um jogo especial (com Megaram gravados em cartucho e usando junto com o cartucho Megaram e no slot 'b' o jogo gravado no cartucho?

5 — Um disco de 5 1/4 tem uma capacidade de 360K ou 720K, qual seria a capacidade de uma fita cassete em um lado e nos dois lados?

MARCELO DE ANDRADE PASSOS
R. ARISTIDES N. CARVALHO, 367
CENTRO.
29200 — GUARAPARI — ES

Para evitar confusão, as respostas estão enumeradas, como se seguem:

1 — Qualquer jogo pode ser gravado em fita. Entretanto, alguns jogos se tornam inviáveis pelo tamanho, ou então, não há interesse por parte do fabricante, já que um programa em fita é mais vulnerável à cópia ilegal.

2 — O MSX pode endereçar, teoricamente, 1 Mbyte, divididos através de seus 4 slots. Desta forma, cada cartucho acessa, no máximo, 256 Kytes.

3 — Sim, não só estes, mas qualquer programa pode ser gravado em cartucho.

4 — A mistura de cartuchos no slot é possível, mas nem sempre há garantia que funcionarão.

5 — A capacidade de um disquete de 5 1/4 pode variar de 18K até 1,4 Mbytes, dependendo do drive. O padrão usado no MSX nos dá 360 Kbytes no total, se usado um disco de dupla face. Como no disco, a capacidade de um fita varia, dependendo entre outros fatores, do tempo de duração e, principalmente, da taxa de transmissão dos dados.

6 — Por fim, um cartucho pode ser gravado e regravado quantas vezes forem necessárias, bastando dispor do equipamento necessário para tal.

MAIS DICAS

Em primeiro lugar, parabéns pelo sucesso da revista.

Mando uma relação de dicas para jogos, pois já vi que muitos leitores se interessam.

Bosconian

```
10 BLOAD"BOSCO1":POKE &H92B-B,250
20 DEFUSR=PEEK (&HFCC0)
*256+PEEK (&HFCBF) :A=USR (0)
30 BLOAD "BOSCO2", R
```

Knight Mare

```
10 BOLAD "KMARE1":PO-KE&H91CA,0
20 DEFUSR=PEEK (&HFC-C0)*256+PEEK(&HFCBF)
30 A=USR (0): BLOAD "",R
```

Zanac

```
10 BLOAD"ZANAC1":POKE&H9654,0
20 DEFUSR=&HD000:A=USR (0):BLOAD"ZANAC2",R
```

Camelot Warriors

```
10 BLOAD"CAMEL1",R:BLOA-D"CAMEL2"
20 POKE &HC55C,250:POKE &HC7B7, 250
30 DEFUSR=PEEK (&HFCC0)
*256+PEEK (&HFCBF) :A=USR (0)
```

Goonies

```
10 BLOAD"GONIE1":POKE &H9261,201
20 DEFUSR=PEEK (&HFCC0)
*256+PEEK (&HFCBF)
30 A=USR (0) : BLOAD "GONIE2", R
```

The Castle

```
10 BLOAD "CASTLE1"POKE &H9D53,250
20 DEFUSR=&HD000:A=USR (0):
BLOAD"CASTLE2",R
```

Stone of Wisdom

```
10 BLOAD "STONE1", R:BLOAD"S-TONE2":POKE &H9327,60
20 DEFUSR=PEEK (&HFCC0)
*256+PEEK (&HFCBF) :A=USR(0)
```

Haunted House

```
10 OPEN "HAUNTED .004"
AS#1:LEN=1
20 FIELD#1,1 AS A$
```

MSXDEBUG E SCREEN IV

Alguns leitores têm, ainda, algumas dúvidas quanto aos projetos SCREEN IV e MSXDEBUG. Mais uma vez, lembramos que os programas só funcionarão se todos os passos forem seguidos à risca. Não basta digitar os blocos, verificar a soma e pronto. Também é necessário seguir o que o texto determina, colocando os dados corretos nos lugares certos. Aliás, é nesta parte que as coisas principais são descritas, principalmente pelo nível de detalhe e importância envolvidos. Os blocos contém apenas rotinas isoladas, devendo ser ligadas através dos dados acrescentados em cada parte.

Todos que digitaram o programa BASIC que cria o MSXDEBUG e ainda recebem a mensagem ERRO NO PROGRAMA devem verificar se a parte montadora está certa. Se estiver, verifique os dados, procurando distinguir caracteres semelhantes como o 8 o B. Não esqueçam que muitos leitores passaram por esta etapa sem problemas.

Ao leitor José Humberto Fregnani, aconselhamos não utilizar artifícios daquele tipo como relatado em sua carta. O erro a que se refere, foi sanado num artigo especialmente elaborado para este fim, na CPU número 10. O fato da troca aleatória daquele valor ter funcionado, não garante que o funcionamento geral do programa não foi afetado. Um programa em linguagem de máquina não deve ser alterado a menos que o programador esteja totalmente seguro daquilo que estiver fazendo. Caso tenha alguma dúvida, não hesite em nos escrever para pedir um esclarecimento, ou mesmo sugerir uma mudança. Espero que isto seja suficiente para convencer a você e aos demais leitores a abandonar idéias como essa.

Acreditamos que suas dúvidas quanto ao comando DSAVE já tenham sido esclarecidas. No número 12 da revista, o artigo do MSXDEBUG foi destinado exclusivamente à discussão dos problemas deste tipo e como resolvê-los. O fato do programa HERO ter sido aceito, é devido o modo de gravação, como explicado naquele artigo.

No número 11, está publicado o bloco inicial do conjunto de chamadas para as rotinas do SCREEN IV. Os leitores que estiverem tendo problemas com o funcionamento do programa, como não conseguir ativar a tela 4 ou não obter o comando PRINT ou WIDTH nesta tela devem rever esta parte. Os valores enviados contendo a soma dos diversos blocos não puderam ser usados a contento, uma vez que a parte inicial do seu programa certamente está diferente.

Aos leitores que já começaram a usar o SCREEN IV, como o leitor Waldir Alves Bezerra, e têm problemas com alguns comandos, no caso o INPUT, aconselhamos aguardar a conclusão do programa. Na quinta parte do projeto, todo ambiente BASIC estará pronto. Antes de implementada esta parte, não é possível manipular arquivos seqüenciais ou randômicos. A propósito, gostaria de saber que caracteres apareciam ilegíveis no vídeo, pois isso realmente não pode ser considerado normal.

As sugestões foram anotadas e a vinda de um desassemblador, como pode perceber, já está disponível.

Sem mais para o momento.

SÉRGIO DURIC CALHEIROS

SOLUÇÃO E DÚVIDA

Estou seguindo, a partir da revista CPU número 8, o projeto Screen IV.

Digitei os 2 blocos publicados com o MSXDEBUG. Executei o programa, entrei o comando "SCREEN 4", a tela de alta resolução foi ativada e, após alguns segundos, o modo texto foi reativado, conforme foi dito.

Na revista número 9 foi publicado o módulo de impressão.

Também foi dito para colocar os bytes FDA4H em 4168H. Eis o erro. Vários leitores escreveram à revista dizendo que não conseguiram imprimir caracteres na nova tela, porque os bytes FDA4H não devem estar em 4168H e sim em 4169H. E o endereço do gancho do comando "WIDTH" deve ficar em 4167H. Após esta correção, o programa funcionou corretamente. Todas as listagens dadas nas revistas números 9 e 10 funcionaram perfeitamente.

Agora, tenho uma pergunta a fazer, com relação ao MSXDEBUG: por que se preenchermos a área de memória a partir de D000H haverá uma queda do sistema?

JORGE DANIEL AGUILAR SOLT
RUA CABO VERDE, 290/174
VILA OLÍMPIA
04550 — SÃO PAULO — SP

Realmente, alguns endereços do SCREEN IV foram publicados com erro, mas acreditamos que, a esta altura do campeonato, todos os usuários do SCREEN IV que passaram por este problema já devem tê-lo solucionado através das listagens que foram publicadas nos números anteriores.

A queda do sistema ocorre porque, quando o MSXDEBUG está na memória, há um interfaceamento

```
30 LSET A$ = CHR$(250)
40 PUT 1,2349: CLOSE

50 REM VOLTE AO DOS E CHAME O
PROGRAMA
```

DANIEL CICERO GONDRAN
R AUGUSTO F DE MORAES 617
SOCORRO
04763 — SÃO PAULO — SP

MEGARAM

Escrevo para a CPU, pela primeira vez, e fiz esta carta para responder à crítica a respeito da Megaram, do Sr. Célio, publicada na CPU número 11.

Depois de ler a carta do Sr. Célio, procurei saber um pouco mais sobre o cartucho Megaram que possuia.

Concordo que para a Megaram são praticamente inexistente essas aplicações profissionais, exceto que podemos usá-las como um Disk drive, ou seja, como Ram Disk.

Creio que você já deve ter ouvido falar a respeito do Megaram Disk plus, da DDX, que permite sua utilização como um Disk drive, sendo muito útil, pois acelera o processamento de dados de programas que constantemente aciona este periférico, como o Page Maker, dBase II, Wordstar, etc.

Porém, o custo elevado do produto nos faz querer bem mais do que acelerar o processamento de programas.

Pesquisando em revistas, achei apenas 2 aplicativos que utilizam a expansão Megaram e ambos só funcionam no MSX 2, só sendo encontrados no Japão. São eles: MSX Write, da ASCII Corporation e o Hal Note da Hal Laboratories.

O pior é que se estes programas chegassem ao Brasil, necessitariam de uma expansão de 5 Megarans para rodar.

Porém, para os fanáticos por jogos, que não são poucos, a expansão Megaram é muito útil, pois utilizada em conjunto com um MSX 2, torna o micro um verdadeiro fliperama.

Qualquer dúvida sobre a existência dos programas japoneses que menciono acima, é só consultar a MSX Magazine de dezembro de 1986.

Para os fanáticos, aí vai uma dica:

No jogo Gradius II (Nemesis II), tecle F1 e escreva as seguintes senhas:

NEMESIS (passa de fase)
LARS 18TH (todas as armas)
METALION (imunidade)

RODNEY MULLER
RUA HERMES DUTRA, 456
12220 — SÃO JOSE DOS CAMPOS — SP

ERRO DE DISCO

Na edição número 11, li a correspondência do Sr. Célio A. F. da Silva e constatei que sua situação se assemelha à minha e queria compará-las.

Um amigo me emprestou um drive para gravar uns programas para ele, de fita para disco. Gravei uns cinco e comecei a ter problemas. O primeiro deles foi com o Batman, que não consegui transferir para disco até hoje.

Depois tive problemas com o Vampire mas, depois de ter feito algumas mudanças, consegui.

Ainda tinha cerca de 27 Kb livres no disco e tentei copiar mais um, o Dizzy Ball. Tudo corria bem até o último bloco, quando, após o término da gravação, obtive a mensagem de erro no disco. Tentei de tudo para ver novamente os programas, sem obter sucesso. Perdi todos os programas do disco.

Gostaria de ajuda, pois está nos meus planos comprar uma unidade de disco e não queria ter os mesmos problemas. Será que o problema é com a interface? Se for, qual à aconselhável?

Aproveito a oportunidade para agradecer a André L. Anciães, Marcos R. Tavares e Eduardo R. Tavares, por terem encontrado os pokes de vida infinita para o El Mundo Perdido. Tentei várias vezes, mas não consegui e mando a dica do jogo Venom.

```
10 COLOR 10,1,1:SCREEN2:OPEN"GRP:"AS#1:PRINT#1,"OLISOFT—VENOM—CPU"
20 BLOAD"VENOM1",R:BLOAD"VENOM2",R:BLOAD"VENOM3"
30 POKE &HB555,0:POKE &HB556,0:POKE &HB557,0:POKE &HB7ED,0:POKE &HB7EA,0:POKE &HB7EB,0:DEFUSR = &H8700
:A = USR(0)
40 BLOAD "VENOM4",R:BLOAD"VENOM5",R
50 REM
60 REM
```

Estes pokes acima fazem com que, quando passarmos de uma tela para outra, se voltarmos, apareceremos no outro lado da tela, a direita.

ARLINDO M. DE OLIVEIRA
ESTRADA DOS BANDEIRANTES, 7993
BLOCO 07 APTº 406
JACAREPAGUA
22780 — RIO DE JANEIRO — RJ

Acreditamos que o problema que aconteceu consigo tenha sido por causa da interface de drive.

A melhor forma de comprar um equipamento é obter as maiores informações possiveis sobre o fabricante, rede de assistência técnica, prazo de garantia e, principalmente, consultar outros usuários que, melhor que ninguém, poderão informar se estão satisfeitos com o produto que possuem. Quanto mais criteriosa for a pesquisa que realizar, maiores serão as chances de adquirir um produto que o satisfaça e que atenda às suas necessidades como usuário.

INTERFACE

Possuo um computador MSX 1.1, um drive e uma impressora. Inicialmente, gostaria de parabenizá-los pela excelente revista, que a cada mês se torna mais interessante. Comecei com a número 9 e pretendo prosseguir com a coleção. A única coisa que acho ruim é o preço, mas temos que nos conformar. Gostaria que vocês me tirassem algumas dúvidas, as quais relaciono abaixo.

Possuo um drive DDX e a sua interface não acessa os arquivos do disco através de portas ((0D0H-0D3H) e sim por memória (07FC8H-07FCCH e 0BFC8H-0BFCCH do slot da interface), fazendo com que alguns arquivos (Robocop, Graphic View, Silent Shadow, Elite, etc) não entrem. Como devo proceder, para que os mesmos funcionem.

Pediria que vocês me ensinassem como fazer uma música em Assembler, mas não naquele estilo de "encher" o registrador HL com notinhas do tipo "CDEFGAB" e depois da um CALL &H73E5, mas sim do jeito que é feito nos jogos e em forma de mneumônicos, explicados passo a passo.

ALEXANDRE TAVARES V DIAS
RUA RIO DE JANEIRO, 139
CAMPO GRANDE
11075 — SANTOS — SP

A interface por endereçamento de memória é a que segue o padrão MSX e a que faz o acesso por portas está fora do padrão. Infelizmente, a primeira interface que foi lançada no Brasil foi a que fazia o acesso por portas e esse sistema foi utilizado por várias empresas. Com isso, vários programas foram adaptados para este tipo de Interface, pois na época era a mais utilizada.

Hoje em dia já temos vários fabricantes com interfaces para drive dentro do padrão, como é o caso da DDX, Gradiente, Sharp e outras e acreditamos que em breve já existiram versões desses softwares que rodem nestas interfaces.

Anotamos suas sugestões e, nos próximos números de CPU, abordaremos os assuntos que nos propõe.

BIBLIOTECAS EM PASCAL

Parte I

FREDERICO DOS SANTOS LIPORACE

É cada vez maior o interesse de programadores sobre a linguagem Pascal, sendo, inclusive, por muitos considerada a "linguagem da moda". Mais do que a própria linguagem, a razão do sucesso do Pascal entre os usuários do MSX é a existência do excelente pacote Turbo-Pascal da Borland, que junta num mesmo programa um poderoso editor e um bom compilador, o que possibilita ao usuário uma facilidade de edição/manutenção de programas que não existe em nenhum outro compilador vindo do CP/M.

Infelizmente, grande parte das pessoas que iniciam o aprendizado da linguagem, desistem ainda cedo, devido à falta generalizada de informações sobre a versão especifica para CP/M, além do fato de não existirem funções internas de acessos gráficos, som e outras potencialidades do MSX.

Já foram publicados, aqui na própria CPU e em outras revistas especializadas, bons artigos de autores diversos que tratavam desse problema. No entanto, os pacotes publicados eram "fechados", ou seja, ficava muito dificil implementar a partir das funções apresentadas outras que fossem consideradas úteis pelo programador.

É neste ponto que a série iniciada nesse número se diferencia das demais. Todas as novas funções apresentadas serão comentadas com a maior didática possivel, possibilitando, assim, implementação de novas funções ao gosto do usuário. Além disso, serão apresentadas não somente as rotinas básicas de manipulação de video e do PSG, mas também procedures de Windows, Impressão de tela gráfica, manipulação do set de caracteres e interrupções, etc., que no final formarão verdadeiras bibliotecas que poderão ser utilizadas para tornar a programação em Pascal muito mais agradável e acessivel.

É bom ressaltar aqui que a série não tem como objetivo ensinar a programar em Pascal, assim como o entendimento das rotinas apresentadas exige um pequeno conhecimento de Assembler e da arquitetura do padrão MSX. Isso, no entanto, não deve tirar o interesse de quem não possua esses conhecimentos, uma vez que as rotinas poderão ser usadas sem problemas por qualquer pessoa que programe em Pascal.

Juntamente com as bibliotecas, serão publicados artigos sobre comandos especificos da versão para CP/M, que, justamente por serem exclusivos, não são apresentados com detalhes pela literatura especializada, que é voltada exclusivamente para a linha PC.

O coração das rotinas apresentadas nesse primeiro bloco e também das que serão apresentadas no futuro é o procedimento CLPRIM. É através dele que é feita toda a comunicação entre o programa Pascal, a BIOS e o próprio interpretador BASIC. O funcionamento detalhado dessa rotina será explicado no artigo que tratar da instrução INLINE. Por enquanto, basta saber que essa instrução permite que sejam inseridas, dentro do programa em Pascal, rotinas em Linguagem de Máquina.

O mais importante é que fique claro como é feita a passagem de dados entre a BIOS e o programa Pascal. Ela é feita através da leitura e/ou escrita em variáveis definidas com o mesmo nome dos registradores do Z-80. Uma vez em posse dos dados necessários, CLPRIM transfere o controle para a BIOS e depois o devolve para o Pascal. Como um exemplo vale mais que 100 palavras, vamos direto a ele.

Vamos supor que houvesse a necessidade de se ler o conteúdo de um endereço da VRAM em determinado programa. O livro vermelho nos diz que existe uma função na BIOS que faz exatamente isso, e indica qual o seu endereço e os parâmetros a serem passados. São dados, portanto:

Nome: RDVRM
Endereço: 004AH
Entrada: HL = Endereço a ser lido
Saida: A = Byte lido

A função em Pascal ficaria, então, escrita da seguinte maneira:

```
Function Vpeek (x:integer):byte;
  begin
     hl: = x;{Transfere o valor de X }
    { para o "par de registradores" }
    { HL }
    Clprim ($004A);
    { O argumento de Clprim é o endereço da rotina   }
    Vpeek: = a{ Valor de retorno está no "registro" A
  end;
```

Outro exemplo: Rotina para transferência de bloco da Vram para a Ram:

Nome: LDIRMV
Endereço: 0059H
Entradas: BC = Tamanho
DE = Endereço da Ram
HL = Endereço da Vram

```
Procedura Vram — Mem (x,y,z: integer);
  begin
    de: = x; { Transfere dados }
    hl: = y;
    bc: = z;
    Clprim ($0059) ;
  end;
```

Uma vez entendido o processo utilizado, não existirão

maiores dificuldades para implementação de novas comandos.

A descrição das comandos apresentados neste 1º bloco é a seguinte:

• Beep: Gera um BEEP.
• FillVram (x, y, z: integer);
Preenche área da Vram com início em y e de tamanho z com byte x.
• Stick (x);
Lê status da joystick x.
• Trig (x);
Lê status do gatilho x.
• GraphMode (x);
Seleciona Screen (x).
• Colar (x, y, z);
Define x como cor de frente, y como cor de fundo e z cama cor de borda.
• Vpeek (x);
Lê conteúdo do endereço x da Vram.
• Vpoke (x, y);
Escreve no endereço x da Vram a byte y.
• Pause: Espera uma tecla ser pressionada.
• LimpSpr: Limpa área de sprites.
• Vr — Mem (x, y, z);
Copia blaco de tamanho z da Vram, a partir do endereço y para a endereço x da Ram.
• Mem-Vr (x, y, z);
Capia bloco de tamanho z a partir do endereço y do Ram para o endereço x da Vram.

• PutSprite (x, y, z, w);
Calaca sprite nº x na posição (y,z) com a car w.
• CreateSpr1 (x,y);
Define sprite y (8x8, tipo 1) e a associa ao número x. Note que y é da tipo Sprite1, ou seja, um array de 8 bytes.
• CreateSpr2 (x, y);
Idem ao anterior, só que é usado para definir sprites de 16x16, y é da tipo Sprite2.
• Size (x);
Define o tamanho do sprite que será utilizado. Se x = 0 as sprites serão apresentados com 8x8 pixels (tipo 1) e se x = 1 serão apresentados com 16x16 pixels (tipo 2).
• Magnitude (x);
Especifica se o sprite será exibido em tamanho ampliado (x = 1) ou normal (x = 0).

O tipo dos parâmetros e dos resultados das funções pode ser identificado através da própria listagem.

Salve o bloco com o nome CLPRIN.INC. Rode o program EXEMPLO, que demonstra e comenta o potencial dos funções de manipulação de sprites implementadas. No mês que vem, completaremos os rotinas gráficas e acessaremos a interpretador BASIC sem correr o risco de abortar o programa, coso haja algum tipo de erro.

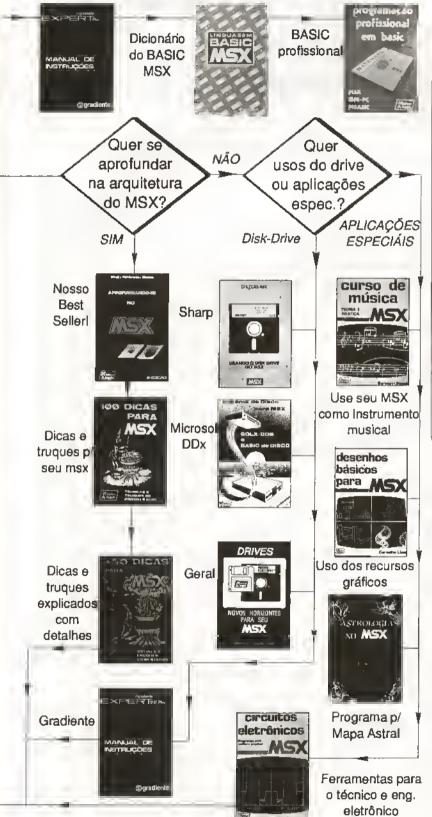
Para terminar, é importante que exista a participação dos leitores, enviando sugestões, críticas, etc. aqui para CPU. Essa participação determinará a prioridade e a forma com que as assuntos serão tratados no decorrer da série. ■

CLPRIN.INC

```
A>type clprim.inc
type
  caracter=array [1..8] of byte;
  sprite1=caracter;
  sprite2=srray [1..32] of byte;
var
  a:byte;              { "Registradores" }
  hl,bc,de:integer; { do Z-80           }
  KeyBuf:array [1..11] of byte;
Procedure Clprim (ix:integer);
  begin
    InLine ($F3/
            $3A/A/
            $2A/HL/
            $ED/$4B/BC/
            $ED/$5B/DE/
            $DD/$2A/IX/
            $CD/*+18/
            $32/A/
            $22/HL/
            $ED/$43/BC/
            $ED/$53/DE/
            $FB/
            $C9/
            $08/
            $DB/$A8/
            $F5/
            $E6/$F0/
            $ED/$73/$9A/$F3/
            $C3/$F38C
           )
end;
```

LIVROS DA ALEPH:
Você tem um micro MSX?
NÃO
Você não sabe o que está perdendo!
SIM
Logo para crianças
HOTLOGO
Resumo do Logo
COMO USAR SEU HOTBIT
Logo
Quer aprender alguma linguagem ou quer manual de software?
Linguagem
Qual linguagem?
BASIC
Você quer curso de Basic
NÃO
SIM
BASIC para Iniciantes
curso de basic
Você é criança?
NÃO
SIM
Quer coleção de programas?
NÃO
SIM
Manuais de Software
Linguagem de Máquina
Banco de Dados
HOTDATA
Assembly Z80 para principiantes
linguagem de máquina
Todos os códigos da L.M. p/ Z80
basic para crianças
Introdução ao BASIC p/ crianças
Editor de textos
HOTWORD
Aplicações do Assembly p/ MSX
Veja Também o Logo!
Programas variados
COLEÇÃO DE PROGRAMAS PARA MSX
Planilha de cálculos
HOTPLAN
Programas em Basic
Resumo de Basic Logo Hotword Hotdata Hotplan
COMO USAR SEU HOTBIT
Programas em Basic e L.M.
COMO RECEBER OS LIVROS EM SUA CASA!
Os livros da ALEPH podem ser encontrados em livrarias, magazines e lojas de computação. Se preferir, para sua maior comodidade e para aproveitar nossa promoção especial para os leitores da CPU, faça seu pedido diretamente. Antes mesmo de ligar seu MSX, mande o pedido ao lado para:
ALEPH P.A.P - Av Dr. Luis Migliano 1110 cj-301 - São Paulo - SP - CEP-05711
e os livros serão enviados imediatamente por porte registrado: as despesas de correio correm por nossa conta e você terá um desconto especial!
Maiores informações: (011) 843-3202

O MELHOR SOFTWARE PARA SEU MSX!

Esta oferta respeita o Código de Proteção ao Consumidor da ABEMD Associação Brasileira de Marketing Direto (011) 826.3458

LISTA DE PREÇOS EM **REAL LIVRO**¤	
HOTDATA	14,7
HOTWORD	13,2
HOTPLAN	14,7
COMO USAR SEU HOTBIT	22,5
LINGUAGEM DE MÁQUINA MSX	21,4
PROGRAMAÇÃO AVANÇADA EM MSX	25,0
TABELA DE MNEMÔNICOS Z80	4,5
HOTLOGO	14,9
BASIC PARA CRIANÇAS	19,2
CURSO DE BASIC VOL.I	22,2
PROGRAMAÇÃO PROFISSIONAL EM BASIC	28,5
EXPERT PLUS-M.DE INSTRUÇÕES	29,5
LINGUAGEM BASIC MSX	31,5
COLEÇÃO DE PROGRAMAS VOL.1	20,0
COLEÇÃO DE PROGRAMAS VOL.2	21,6
JOGOS DE HABILIDADE	14,9
APROFUNDANDO-SE NO MSX	27,5
100 DICAS PARA MSX	27,5
+50 DICAS PARA MSX	29,5
USANDO O DISK DRIVE NO MSX	27,5
SISTEMA DE DISCO PARA MSX	20,4
DRIVES-NOVOS HORIZONTES PARA SEU MSX	22,2
EXPERT DD PLUS-M.DE INSTRUÇÕES	28,0
CURSO DE MÚSICA MSX	20,4
DESENHOS BÁSICOS NO MSX	18,3
ASTROLOGIA NO MSX	24,9
CIRCUITOS ELETRÔNICOS MSX	27,5

¤ **REAL LIVRO** é a unidade monetária estabelecida pela CÂMERA BRASILEIRA DO LIVRO, cuja correspondência atual é

1 REAL LIVRO = NCz$ 70,00

SIM, quero receber pelo correio por porte registrado os livros discriminados abaixo:

Título	Preço
TOTAL EM REAL LIVRO	
VALOR DO REAL 70,00	
TOTAL EM NCz$	
Desconto 15% (até 30/03/90)	
VALOR A PAGAR —NCz$	

Estou enviando um cheque cruzado nominal à ALEPH P.A.P. Ltda do banco nº__________agencia nº_____ no valor de nCz$__________para receber os livros indicados acima sem outras despesas de minha parte.

NOME: ____________________

END: ____________________

CEP:______CIDADE:______________UF:___

Tenho um drive de (_)5 1/4" (_)3 1/2" e quero receber um catálogo dos kits (livro+ disco).

PASLIB1. INC

```
Procedure Beep;
  begin
    Clprim ($ØØCØ)
  end;
Procedure FillVram (x:byte;y,z:integer);
  begin
    a:=x;
    hl:=y;
    bc:=z;
    Clprim ($ØØ56)
  end;
Function Stick (x:byte):byte;
   begin
     a:=x;
     Clprim ($ØØD5);
     Stick:=a
   end;
Function Trig (x:byte):boolean;
   begin
     a:=x;
     Clprim ($ØØD8);
     Trig:=(a=255)
   end;
Procedure GraphMode (x:byte);
   begin
     a:=x;
     Clprim ($ØØ5F);
     if x=Ø then mem[$F3BØ]:=4Ø
            else if x=1 then mem[$F3BØ]:=32;
     mem [$FCB3]:=Ø;
     mem [$FCB5]:=Ø
   end;
Procedure Color (x,y,z:byte);
   begin
     mem [$F3E9]:=x;
     mem [$F3EA]:=y;
     mem [$F3EB]:=z;
     Clprim ($ØØ62)
   end;
Function Vpeek (x:integer):byte;
   begin
     hl:=x;
     Clprim ($ØØ4A);
     vpeek:=a
   end;
Procedure Vpoke (x:integer;b:byte);
   begin
     a:=b;
     hl:=x;
     inline ( $F3/
              $2A/HL/
              $99ØE/
              $69ED/
```

```
                    $F4CB/
                    $61ED/
                    $3A/A/
                    $98D3/
                    $FB
                  )
    end;
  Procedure Pause;
    begin
      Clprim ($009F)
    end;
  Procedure LimpSpr;
    begin
      Clprim ($0069)
    end;
  Procedure Vr_mem (a,b,c:integer);
    begin
      bc:=c;
      de:=a;
      hl:=b;
      Clprim ($0059)
    end;
  Procedure Mem_vr (a,b,c:integer);
    begin
      bc:=c;
      de:=a;
      hl:=b;
      Clprim ($005C)
    end;
  Procedure PutSprite (b,d,c,e:byte);
    var
       aux:integer;
    begin
      a:=b;
```

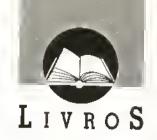
```
        Clprim ($ØØ87);
        aux:=hl;
        vpoke (aux,c);
        vpoke (aux+1,d);
        vpoke (aux+3,e)
      end;
   Function GetPat (x:byte):integer;
      begin
        a:=x;
        Clprim ($ØØ84);
        GetPat:=hl
      end;
   Procedure CreateSpr1 (x:byte;y:sprite1);
      var
        aux,aux1:integer;
      begin
        aux1:=GetPat (x);
        for aux:=Ø to 7 do
          Vpoke (aux1+aux,y[aux+1])
      end;
   Procedure CreateSpr2 (x:byte;y:sprite2);
      var
        aux,aux1:integer;
      begin
        aux1:=GetPat (x);
        for aux:=Ø to 31 do
          Vpoke (aux1+aux,y[aux+1])
      end;
   Procedure Size (x:byte);
      begin
        bc:=mem [$F3EØ]*256+1;
        if x=1 then bc:=bc or 512
               else bc:=bc and (not(512));
        Clprim ($ØØ47)
      end;
   Procedure Magnitude (x:byte);
      begin
        bc:=mem [$F3EØ]*256+1;
        if x=1 then bc:=bc or 256
               else bc:=bc and (not(256));
        Clprim ($ØØ47)
      end;
```

EXEMPLO. PAS

```
{$i clprim.inc}  { Inclui Clprim }
{$i paslib1.inc} { Inclui Biblioteca1 }

const
  Raio=70;
  Pi=3.1415;
  def=100;
  s1:sprite2=(3,15,31,63,63,127,127,127,
              127,127,63,63,31,15,3,0,
              192,240,248,252,252,254,254,254,
              254,254,252,252,248,240,192,0);
  { O tipo Sprite2 e um Sprite de 16x16 }
  { pixels, definidos por 32 bytes como }
  { no BASIC }

var
  PosX,PosY:array [1..def] of byte;
  ang,da:real;
  cont,vezes:byte;

begin
  da:=(2*Pi/def);
  ang:=0;
  for hl:=1 to def do begin
    PosX[hl]:=trunc (Raio*Cos(ang));
    PosY[hl]:=trunc (Raio*Sin(ang));
    ang:=ang+da
  end;
  { Note que como hl foi definida  como }
  { uma variavel global, ela  pode  ser }
  { usada sem problemas em lugares onde }
  { Clprim nao seja acessada.           }
  GraphMode (1); { Screen (1) }
  Size (1);
  Magnitude (1);
  { Experimente Magnitude (0) }
  LimpSpr;
  for cont:=0 to 5 do CreateSpr2 (cont,s1);
  { Definicao dos Sprites }
  for vezes:=1 to 10 do
    for cont:=1 to def do begin
      PutSprite (0,100+PosX[cont],80+PosY[cont],3);
      PutSprite (1,100-PosX[cont],80-PosY[cont],8);
      PutSprite (2,100+PosX[cont],80,10);
      PutSprite (4,100,80+PosY[cont],15);
      delay (8);
      { Experimente diminuir o argumento de Delay }
    end;
  GraphMode (0);
end.
```

DESENHOS BÁSICOS PARA MSX

DIVINO C. R. LEITÃO

A Editora Aleph colocou, recentemente, no mercado, um novo livro para MSX, desta vez dirigido aos apreciadores de artes gráficas, de autoria de Carvalho Lima, que inicia brilhantemente sua obra com uma citação de Isaac Asimov — "Eu tento ensinar aos outros porque, seja ou não bem sucedido, eu invariavelmente consigo ensinar a mim mesmo".

Após leitura atenta e execução de algumas rotinas, fiquei algo decepcionado, pois não encontrei grandes novidades. Mas comigo não vale, pois artes gráficas são meu forte no computador e minha decepção foi em virtude de não encontrar algo novo para o meu uso. Porém, identifiquei no decorrer da obra alguns dos meus maiores tormentos da época que ainda engatinhava pelos "pixels" dos micros. Eles estavam lá, explicadinhos, de uma forma que ninguém explicou quando eu estava aprendendo.

No livro, são tratados com a mesma desenvoltura e clareza, desde um simples comando PSET até a tenebrosa trigonometria, com seus senóides e outros fantasmas que assustam os desavisados. Com uma linguagem bem didática e com pontadas de humor, o autor leva os usuários ao interessante mundo dos gráficos. Se não sabe sequer traçar uma linha reta, este livro foi feito para você; para os que já dominam os comandos gráficos do MSX, pouco há a acrescentar.

Sob um selo bem adequado, "série didática", Desenhos Básicos para MSX pode se revelar um bom auxiliar a professores de desenho ou geometria que se interessem em usar o micro como apoio ao ensino. Todos os exemplos são acompanhados de exercícios relativos ao assunto tratado e com resposta no final da obra. Alguns efeitos são bastantes interessantes.

No final do livro, existem algumas tabelas quadriculadas com a representação dos modos de tela screen 1 e 2, e aí vale uma crítica, com relação à cor escolhida para estas tabelas, fundo azul e traços em branco, além de uma péssima visualização, que impedem a cópia das tabelas em xerox, o que seria muito útil aos usuários do livro.

Achei que o screen 3, que possui recursos, como a própria autor diz, misteriosos, continuou misterioso, pois foi pouco explorado pela obra, assim como o fantástico macro-comando DRAW, do qual foram esquecidos alguns recursos, como o comando x, que permite chamar uma sub-rotina DRAW dentro de outra.

CONCLUSÃO

No geral, é uma boa obra, talvez um pouco curta, o que pode ser uma característica da série que representa. Recomendo-a como um complemento ao manual do micro para os iniciantes e também como um estudo de aplicação da geometria no micro MSX.

TUDO PARA MSX

DRIVE

5 1/4"

CANETA ÓTICA

3 1/2

EXPANSOR
DE SLOTS

MODENS

TECLADOS
RS 232
TERMINAL
TELEX PARA MSX
PROGRAMAS
SCREEN IV
MSX BUG
E MUITOS OUTROS

DISKETES

ENTREGA IMEDIATA

ÁGUIA INFORMÁTICA LTDA.

Av. N.S. de Copacabana, 605/804 – Copacabana
Telex: 21 21 717 KPUR – Tel.: (021) 235 3541
Rio de Janeiro – RJ – CEP 22040

DRAWER Sistema Editor

RENATO PAULO DE MELLO

O Drower — Sistema Editor tem por finolidode facilitor o utilizoção do camondo DRAW, crionda um orquivo no formoto BASIC, que poderá ser focilmente implementado em outros progromos, otrovés do MERGE.

O pragromo é dividido em duas partes distintos:
— Telo de Edição
— Menu Principol

Telo de Edição:

Nesto tela, pade-se desenhor livremente com o ouxílio dos teclos de movimentação do cursar, tendo-se á disposição alguns recursos, os quais podem ser acessados através dos teclos de funções.

As opções possíveis sãa:

— RISCAR / NÃO RISCAR — esta apção é conseguido otravés da tecla de função F-1. Elo funciono como uma chave ligo / desligo. O cursor, em farmato de seta, quando representodo cheio, indico que o função está otivo Coso contrária, o cursor será vozodo.

— SELECIONAR CORES — a tecla F-2 mostro o paleto de cores, as quois poderão ser selecionodos com a ouxílio dos teclas do cursor poro cimo e poro boixo. Confirma-se a cor cam < ENTER > .

A cor do borda da telo indico o car em uso.

A teclo <BS> aborto o função.

— GRID — é acessada através da teclo F-3. É desenhado no tela um reticulodo (GRID) 8 X 8. Tem o função liga/desligo.

— APAGAR — pode-se opogor porte de um desenho teclando-se F-4. Será solicitodo o número de pixels o serem apagados. O volor iniciol é iguol o 15. Este volor poderá ser incrementodo ou decrementodo com as teclos do cursar poro cimo e poro baixo. Paro que a operação seja efetuodo, basto pressionarmas <ENTER> .

O número máximo de pixels o serem apagodos em umo operoção é de 255 pixels.

A tecla <ESC> abondono o funçõa.

— MENU PRINCIPAL — é ocessodo com o tecla F-5.

Menu Principal:

Nesto telo, poderemos selecianor a opção desejoda com o auxílio das teclos do cursar para cima e poro boixa confirmondo com <RETURN>.

As opções são:

a) MOSTRAR TELA — mostra a tela que está senda editodo. Paro voltar ao menu, pressione <ENTER>.

b) LISTAR COMANDOS — imprime os comondas no formota BASIC cam numeroção o portir de 50.000. Será solicitoda a escolo desejada, bem camo o sistemo de caordenadas a ser empregodo. Os valores paro o escola devem estar entre 1 o 255 enquonto que para a coordenoda, entre 0 e 3.

Os dadas digitodos podem ser corrigidos com <BS> .

Para monter os valores "defoult", bosto teclarmos <ENTER>.

Após o processamenta, será efetuodo o listogem.

c) GRAVAR NO FORMATO BASIC — grava em disca ou em fita um orquivo de linhos "DRAW" que poderá ser implementodo em outros progromos em BASIC com o MERGE. Será solicitada o soldo o ser utilizoda (disco/fita), a name do progromo, o escola e a sistema de coordenodos. Todos os informoções dados poro o apçãa "b" (Listar comonda) são válidos para esto opção.

Apás o processamenta será executodo o gravação.

d) GRAVAR ARQUIVO — permite grovor em disco ou fito um orquivo da tela de edição. Será pedido o meio de soldo (disco/fito) bem como a nome do arqulvo o ser gravado.

e) LER ARQUIVO — permite recuperar do disca ou fita um orquivo previomente grovodo poro que se posso continuar suo edição. Será pedido o meia de entrada (disco/fito) e a nome do arquivo o ser carregado.

f) EXIBIR DIRETÓRIO DO DISCO — mostro o conteúdo e o espaço disponível do disco no drive "A". Não permite seleção do drive.

g) VOLTAR AO MODO DE EDIÇÃO — permite continuar/ começar o edição de umo telo.

h) APAGAR O QUE FOI FEITO — opogo do memário todo o trabalho reolizodo. Será pedido o confirmoção.

i) VOLTAR AO BASIC — encerro o pragrama.

OBSERVAÇÕES:

1) Sempre antes de umo saido no farmato BASIC (disca, fito ou impressara) é feita o montagem dos comondos. O tempo gasta pora o processomento será proporcional oo tomanho e á complexidade do desenho. O tempo média gasto poro umo tela que utilize toda a espaço dispanível pora edição é de 05 minutos.

Para que o usuário não tenha a impressão de que o micro "congelou", foi implementada uma tela para que se possa ter um acompanhamento visual do processo.

Este processo, uma vez feito, não será mais utilizado até que se volte a editar a tela, que se apague o que foi feito ou, então, um novo arquivo seja carregado.

2) O programa guarda uma área de trabalho suficientemente grande para a elaboração de desenhos com relativo grau de complexidade. Caso este limite seja ultrapassado, será dado um aviso e as únicas teclas que funcionarão serão F-4 (apagar) e F-5 (menu).

Quando um desenho for excessivamente grande e complexo, os limites da rotina de montagem dos comandos poderão ser ultrapassados. Nestes casos (raros), o processamento será interrompido com a mensagem "Muito complexo" e o processamento será desviado para o menu principal. A partir deste momento, todas as funções de saída no formato BASIC serão efetuadas somente até os limites de processamento.

3) O programa não permite a seleção do drive utilizado. Todas as operações que utilizem este meio serão realizadas no drive "A".

4) Na opção Disco, sempre antes do programa efetuar uma gravação, ele verifica se já existe um programa com mesmo nome. Caso já exista um arquivo/programa semelhante, será solicitada a confirmação.

5) Todas a operações de entrada de dados via teclado, podem ser interrompidas com a tecla <ESC>.

6) Os valores "default" adotados pelo programa para a escala e as coordenadas são quatro (04) e zero (00), respectivamente. O meio de saída "default" adotado pelo sistema é o disco.

Para que o programa assuma os valores "default", basta teclarmos <ENTER> em resposta ao dado solicitado.

CONSIDERAÇÕES SOBRE O PROGRAMA

O programa faz uso de uma pequena rotina em linguagem de máquina para criar o GRID, a qual está alocada a partir do endereço &HD000. Além dela, o programa utiliza mais duas rotinas da ROM, que são:

— DEFUSR1 = &H156 — Limpa buffer de teclado.

— DEFUSR2 = &H3E — Reestabelece as teclas de função.

— Foi elaborada uma rotina de tratamento de erros que contorna a maioria das situações. Mesmo que ocorra um erro não definido (tentativa de se trabalhar com um arquivo não padrão, por exemplo), o programa não será finalizado. Será apresentado, nestes casos, a mensagem "ERRO DE PROCESSAMENTO". Portanto, durante a fase de depuração do programa, é aconselhável mudar a linha 3600 para:

3600 SCREEN 0,,1:COLOR 15,4:PRINT "Erro nº", ERR; "na linha"; ERL: LIST.:END

Deste modo, sempre que ocorrer um erro, será dado o número do erro, a linha em que ele ocorreu e já será listada a linha com problemas.

Após o programa estar definitivamente editado, você deverá colocar novamente a linha original.

3600 SCREEN 0:COLOR 15,4:PRINT "Erro de processamento"

— Como o programa desativa as teclas <CTRL+STOP>, é aconselhável a gravação do programa antes de sua execução.

— Buscando a economia de memória, o programa peca pela sua clareza e falta de comentários no interior de sua listagem. Para tentar sanar um pouco o problema, é dada, a seguir, uma tabela de linhas com uma breve descrição.

TABELA DE LINHAS

20 — 260 — Rotina de montagem de comandos.
280 — 310 — Tratamento das interrupções.
330 — 750 — Rotina principal.
770 — 860 — Rotinas auxiliares.
880 — 1010 — F-1 — Risca / não risca.
1030 — 1180 — F-2 — Seleciona cor.
1200 — 1230 — F-3 — Grid.
1250 — 1710 — F-4 — Apagar.
1730 — 1740 — Mostra DRAW.
1760 — 1980 — F-5 — Menu principal.
2000 — 2020 — Mostra tela.
2040 — 2080 — Listar comandos.
2100 — 2190 — Gravar formato BASIC.
2210 — 2230 — Rotinas auxiliares.
2250 — 2470 — Entra escala / coordenadas.
2490 — 2630 — Gravar arquivo.
2650 — 2730 — Teste de existência de arquivo igual.
2750 — 2860 — Ler arquivo.
2880 — 2930 — Seleciona disco/fita.
2950 — 2960 — Mostra diretório.
2970 — 3120 — Entra nome arquivo/programa.
3140 — 3150 — Volta a editar.
3170 — 3220 — Apagar tudo.
3240 — 3250 — Voltar ao BASIC.
3260 — 3280 — Existem comandos?
3300 — 3370 — Rotina saída formato BASIC.
3390 — 3490 — Inicialização.
3510 — 3670 — Rotina de tratamento de erros.
3690 — 3720 — Aviso de limite "estourado".
3740 — 3820 — Abertura.
3830 — 3900 — DATA's L.M. e SPRITES.

Dados do autor:
RENATO PAULO DE MELLO — formado em Processamento de Dados pela UTESC — JOINVILLE (SC)
Autor dos programas SOBRA UM e CRIPTOARITMÉTICA publicado na edição nº 11.

```
1 ' ===================================
2 '
3 '           DRAWER - MSX
4 '         versão 1.0 - 1989
5 '   Autor:Renato Paulo de Mello
6 '         Joinville - SC
7 '
8 ' ===================================
9 '
10 CLEAR8300,&HCFFF:DEFINTA-Z:GOTO280
20 W$="":Y$="":FORI=0TO9:M$(I)="":NEXT
30 CN=14:FC=1:CT=0:V=0:LA=((K*250)+LEN(D
$(K)))\8:TN=0:GOSUB780:LOCATE5,11:PRINT"
*** Aguarde processamento ***"
40 LOCATE13,14:PRINT"!----------!"
50 LOCATE13,15:PRINT"! 0        1 !"
60 LOCATE13,16:PRINT"! !........ !"
70 LOCATE13,17:PRINT"!__________!"
80 FORI=0TOK:FORJ=1TOLEN(D$(I)):X$=MID$(
D$(I),J,1):TN=TN+1
90 IFTN>LATHENCN=CN+1:LOCATECN,16:PRINT"
.!":TN=0:BEEP
100 T=INSTR("urdlefgh",X$)
110 IFT=0THENGOSUB230:GOTO150
120 IFW$=""THENW$=X$:GOTO150
130 IFW$=X$THENCT=CT+1:GOTO150
140 GOSUB170
150 NEXTJ,I:GOSUB170
160 LOCATE22,16:BEEP:PRINT".!":RETURN
170 CT=CT+1:Q$=STR$(CT)
180 IFLEN(M$(V))>230THENV=V+1
190 IFV>9THENV=9:ERROR99
200 IF CT=1 THEN M$(V)=M$(V)+W$:GOTO220
210 M$(V)=M$(V)+W$+RIGHT$(Q$,LEN(Q$)-1)
220 CT=0:W$=X$:RETURN
230 IFX$="b"THENX$=X$+MID$(D$(I),J+1,1):
J=J+1
240 IFW$=""THENM$(V)=M$(V)+X$:RETURN
250 IFCT>0ORW$<>""THENGOSUB170
260 M$(V)=M$(V)+X$:W$="":RETURN
270 '
280 ONSTOPGOSUB430:STOPON
290 ONERRORGOTO3510
300 GOSUB3390
310 ONKEYGOSUB880,1030,1200,1250,1760
320 '
330 IF FLAG=1THENDRAW"M=X;,=Y;C=C;"
340 PUTSPRITENS,(X,Y),CS,NS
350 IF LEN(D$(K))>244THENK=K+1
360 IF K>20THEN K=20:EX=1:GOSUB3690
370 A=STICK(0)
380 ON A GOSUB400,440,490,530,580,620,67
0,710
390 GOTO330
400 IFY=0THENY=0:RETURN
410 Y=Y-1
420 IFFLAG=1THEND$(K)=D$(K)+"u"
430 RETURN
440 IFY=0THENY=0:RETURN
450 IFX=255THENX=255:RETURN
460 Y=Y-1:X=X+1
470 IFFLAG=1THEND$(K)=D$(K)+"e"
480 RETURN
490 IFX=255THENX=255:RETURN
500 X=X+1
510 IFFLAG=1THEND$(K)=D$(K)+"r"
520 RETURN
530 IFY=191THENY=191:RETURN
540 IFX=255THENX=255:RETURN
550 Y=Y+1:X=X+1
560 IFFLAG=1THEND$(K)=D$(K)+"f"
570 RETURN
580 IFY=191THENY=191:RETURN
590 Y=Y+1
600 IFFLAG=1THEND$(K)=D$(K)+"d"
610 RETURN
620 IFY=191THENY=191:RETURN
630 IFX=0THENX=0:RETURN
640 Y=Y+1:X=X-1
650 IFFLAG=1THEND$(K)=D$(K)+"g"
660 RETURN
670 IFX=0THENX=0:RETURN
680 X=X-1
690 IFFLAG=1THEND$(K)=D$(K)+"l"
700 RETURN
710 IFY=0THENY=0:RETURN
720 IFX=0THENX=0:RETURN
730 Y=Y-1:X=X-1
740 IF FLAG=1THEND$(K)=D$(K)+"h"
750 RETURN
760 '
770 L$=STR$(CT):D$(K)=D$(K)+Z$+RIGHT$(L$
,LEN(L$)-1):CT=0:Z$="":RETURN
780 FORI=4TO22:LOCATE0,I:PRINTSPC(38):NE
XT:RETURN
790 LOCATE5,22:PRINT"Tecle <RETURN> para
 continuar"
800 A=USR1(0)
810 A$=INKEY$
820 IFA$<>CHR$(13)THEN810ELSEBEEP:RETURN
830 LOCATE5,21:PRINT"Tecle <RETURN> para
 continuar.":LOCATE5,22:PRINT"Tecle <ESC
>    para cancelar.":A=USR1(0)
840 A$=INKEY$
850 IFA$=CHR$(27)THENBEEP:RETURN1780
860 IFA$<>CHR$(13)THEN840ELSEBEEP:RETURN
870 '
880 PUTSPRITENS,(0,209),0,NS:NS=NSXOR1:P
UTSPRITENS,(X,Y),CS,NS:BEEP
890 IFFLAG=2THENFLAG=1:GOTO990
```

```
900 IFFLAG=1THENFLAG=0:X1=X:Y1=Y:RETURN
910 FLAG=1
920 X0=X-X1:Y0=Y-Y1
930 IFX=X1THEN 960
940 IFX>X1THENZ$="br"ELSEZ$="bl"
950 CT=ABS(X0):GOSUB770
960 IFY=Y1THENRETURN
970 IFY>Y1THENZ$="bd"ELSEZ$="bu"
980 CT=ABS(Y0):GOSUB770:RETURN
990 Z$="bm":CT=X:GOSUB770
1000 Z$=",":CT=Y:GOSUB770
1010 Z$="c":CT=C:GOSUB770:RETURN
1020 '
1030 FORI=1TO5:KEY(I)OFF:NEXT:BEEP:PUTSP
RITE1,(0,209),15,1:L=1:LINE(0,7)-(25,160
),1,BF
1040 FORI=0TO140STEP10:LINE(10,10+I)-(20
,15+I),L,BF:L=L+1:NEXT:LINE(10,10)-(20,1
5),15,B:L=C*10+2:A=USR1(0)
1050 A=STICK(0)
1060 PUTSPRITE0,(17,L),15,0
1070 IFA=1THEN IFL>12THENL=L-10
1080 IFA=5THEN IFL<152THENL=L+10
1090 A$=INKEY$
1100 IFA$=CHR$(27)THENBEEP:GOTO1160
1110 IFA$<>CHR$(13)THEN1050
1120 BEEP:CT=L\10
1130 IFC=CTTHEN 1160ELSEC=CT:Z$="c"
1140 IFD$(0)=""THEN1160
1150 GOSUB770
1160 LINE(0,9)-(25,160),1,BF:COLOR,,C
1170 PUTSPRITE0,(0,209),15,0:GOSUB1730
1180 FORI=1TO5:KEY(I)ON:NEXT:RETURN
1190 '
1200 FORI=1TO5:KEY(I)OFF:NEXT:BEEP:CS=CS
 XOR 11
1210 IF GRID=1THENGRID=0:CLS:GOTO1230
1220 GRID=1
1230 GOSUB1730:FORI=1TO5:KEY(I)ON:NEXT:R
ETURN
1240 '
1250 COLOR,,C:L=15:IFD$(0)=""THENRETURN
1260 FORI=1TO5:KEY(I)OFF:NEXT:BEEP
1270 A=USR1(0):OPEN"grp:"FOROUTPUTAS#1
1280 IF GRID=1THENCN=1 ELSECN=15
1290 IF Y<100THENLA=145 ELSELA=0
1300 LINE(188,8+LA)-(252,41+LA),CN,B
1310 LINE(190,10+LA)-(250,39+LA),CN,BF
1320 COLORCN XOR14:PSET(195,LA+13),CN:PR
INT#1,"Pixels?":PSET(207,LA+25),CN:PRINT
#1,USING"###";L
1330 A$=INKEY$
1340 IF A$<>CHR$(31)ANDA$<>CHR$(30)ANDA$
<>CHR$(13)ANDA$<>CHR$(27)THEN1330
1350 BEEP
1360 IFA$=CHR$(13)THEN 1430
1370 IFA$=CHR$(31)THEN L=L-1
1380 IFA$=CHR$(30)THEN L=L+1
1390 IFA$=CHR$(27)THEN 1690
1400 IFL<1 THEN L=255
1410 IFL>255THENL=1
1420 LINE(205,20+LA)-(240,35+LA),CN,BF:P
SET(207,25+LA),CN:PRINT#1,USING"###";L:G
OTO1330
1430 LINE(190,10+LA)-(250,39+LA),CN,BF:P
SET(195,19+LA),CN:PRINT#1,"Aguarde"
1440 X$="udrlegfhcm":CT=0:CT$="":EX=0
1450 FORI=KTO 0 STEP-1:TA=LEN(D$(I))
1460 FORJ=TA TO2STEP-1:R$=MID$(D$(I),J,1
):T=INSTR(X$,R$)
1470 IF T=9 THENC2=1:GOTO1530
1480 IF T=10THENGOSUB1650:GOTO1660
1490 IF T=0 THEN1530
1500 IF J>1 THENIF MID$(D$(I),J-1,1)="b"
THEN 1530
1510 CT=CT+1
1520 IF CT>LTHEN1540
1530 NEXTJ,I
1540 B$=LEFT$(D$(I),J)
1550 FOR J=ITOK:D$(J)="":NEXT:K=I
1560 D$(K)=B$
1570 IF GRID=0THENCLS
1580 GOSUB1730
1590 X=PEEK(&HFCB3):Y=PEEK(&HFCB5)
1600 IFT=1THEN Y=Y-1:GOTO1630
1610 IFT=5THEN Y=Y-1:X=X+1:GOTO1630
1620 IFT=8THEN Y=Y-1:X=X-1:GOTO1630
1630 IFC2=1THEN CT=C:Z$="c":GOSUB770
1640 X1=X:Y1=Y:PUTSPRITE NS,(X,Y),CS,NS
1650 COLOR,,C:CLOSE:FORI=1TO5:KEY(I)ON:N
EXT:RETURN
1660 ERASED$:DIMD$(20):FLAG=2:NS=1:X=128
:Y=96:K=0
1670 IF GRID=1THEN A=USR0(1) ELSECLS
1680 PUTSPRITE0,(0,209),CS,0:RETURN
1690 IF GRID=1THEN 1710
1700 LINE(188,8+LA)-(252,41+LA),1,BF
1710 GOSUB1730:FORI=1TO5:KEY(I)ON:NEXT:C
LOSE:RETURN
1720 '
1730 IF GRID=1THENA=USR0(0)
1740 DRAW"s4a0":FORI=0TOK:DRAWD$(I):NEXT
:RETURN
1750 '
1760 FORI=1TO5:KEY(I)OFF:NEXT:BEEP
1770 SCREEN0
1780 CLS:COLOR15,4:WIDTH38:L=7
1790 A=USR1(0):LOCATE10,1:PRINT"***  D R
 A W E R  ***":LOCATE0,7
1800 PRINTTAB(7)"Mostrar tela"
```

```
1810 PRINTTAB(7)"Listar comandos"
1820 PRINTTAB(7)"Gravar no formato BASIC
"
1830 PRINTTAB(7)"Gravar arquivo"
1840 PRINTTAB(7)"Ler arquivo"
1850 PRINTTAB(7)"Mostrar diretório do di
sco"
1860 PRINTTAB(7)"Voltar ao modo de ediçã
o"
1870 PRINTTAB(7)"Apagar o que foi feito"
1880 PRINTTAB(7)"Voltar ao Basic"
1890 LOCATE10,22:PRINT"Selecione com <RE
TURN>"
1900 A=STICK(0)
1910 IF A=1THENIFL=7THENL=7ELSEL=L-1
1920 IF A=5THENIFL=15THENL=15ELSEL=L+1
1930 LOCATE5,L:PRINTCHR$(249):LOCATE5,L-
1:PRINT" ":LOCATE5,L+1:PRINT" "
1940 A$=INKEY$
1950 IF A$<>CHR$(13)THEN1900
1960 BEEP
1970 ON L-6 GOSUB 2000,2040,2100,2490,27
50,2950,3140,3170,3240
1980 GOTO1780
1990 '
2000 CLS:MG$="mostrar":NO$="Mostrar tela
":IF D$(0)=""THEN GOSUB3260
2010 COLOR15,1,C:SCREEN2:GOSUB1730
2020 GOSUB800:RETURN1770
2030 '
2040 NO$="Listar Comandos":MG$="listar":
GOSUB3260:GOSUB2250
2050 IF FC=0 THENGOSUB 20
2060 GOSUB780:LOCATE5,11:PRINT"*** Prepa
re a impressora ***"
2070 GOSUB830:N$="lpt:"
2080 MG$="Imprimindo...":GOSUB2230:GOSUB
3300:RETURN
2090 '
2100 NO$="Gravação formato BASIC":MG$="g
ravar":GOSUB3260
2110 GOSUB2880
2120 IFA$="D"THENNO$=".BAS"ELSENO$=""
2130 MG$="programa":GOSUB2970:GOSUB2250
2140 IF FC=0THENGOSUB20
2150 IF P$<>"A:"THEN MG$="gravação":GOSU
B2210:GOTO2190
2160 GOSUB2220
2170 XL=1:GOSUB2650
2180 IFA$="N"THENGOSUB780:TN=8:GOTO2130
2190 GOSUB3300:RETURN
2200 '
2210 GOSUB780:LOCATE0,11:PRINT"*** Prepa
re gravador para ";MG$;" ***":GOSUB830
2220 IF L=11THENMG$="Carregando..."ELSE
MG$="Gravando..."
2230 GOSUB780:LOCATE5,11:PRINT"*** Aguar
de! ";MG$;" ***":RETURN
2240 '
2250 LOCATE5,22:PRINT"<ESC> Retorna ao m
enu principal":LOCATE5,12:PRINT"Escala
 (=004)  : ___"
2260 TN=3:CN=12:LA=255:LB=1:X$="___":GOS
UB2330
2270 IFS$=""THENS$="004":LOCATE24,12:PRI
NTS$
2280 S=VAL(S$):LOCATE5,14:PRINT"Coordena
da (=0)  : _"
2290 TN=1:CN=14:LA=3:LB=0:X$="_":GOSUB23
30
2300 IFS$=""THENS$="0":LOCATE24,14:PRINT
S$
2310 CO=VAL(S$):RETURN
2320 '
2330 S$="":LOCATE24,CN
2340 A$=INKEY$:IFA$=""THEN2340
2350 G=ASC(A$):BEEP
2360 IF (G<48 OR G>57) AND G<>13 AND G<>
8 AND G<>27 THEN2340
2370 IF G=27THENRETURN 1780
2380 IF G=8 AND S$=""THEN2330
2390 IF G=8 THENS$=LEFT$(S$,LEN(S$)-1):L
OCATE24,CN:PRINTS$"_";:GOTO2340
2400 IF G=13 AND S$=""THENRETURN
2410 IF G=13 THEN2460
2420 IF LEN(S$)=TN THEN2340
2430 S$=S$+A$
2440 LOCATE23+LEN(S$),CN:PRINTA$;
2450 GOTO2340
2460 IF VAL(S$)<LB ORVAL(S$)>LA ORLEN(S$
)<>TN THENLOCATE24,CN:PRINTX$:GOTO2330
2470 RETURN
2480 '
2490 NO$="Gravar arquivo":MG$="gravar":G
OSUB3260:GOSUB2880
2500 IF A$="D"THENNO$=".DRW"ELSENO$=""
2510 MG$="arquivo ":GOSUB2970
2520 IF P$<>"A:"THEN MG$="gravação":GOSU
B2210:GOTO2560
2530 GOSUB2220
2540 XL=1:GOSUB2650
2550 IFA$="N"THENGOSUB780:TN=8:GOTO2510
2560 OPEN N$ FOROUTPUTAS#1
2570 IF FLAG=1THEN GX=X:GY=Y ELSEGX=X1:G
Y=Y1
2580 PRINT#1,GX,GY,C,K
2590 FORI=0TOK
2600 T=INSTR(D$(I),","):X$=D$(I)
2610 IF T>0THEN MID$(X$,T,1)="*"
2620 PRINT#1,X$:NEXT
```

```
2630 CLOSE:GOSUB3370:RETURN
2640
2650 OPEN N$ FORAPPENDAS#1
2660 IF LOF(1)>1THENCLOSE:GOTO2680
2670 CLOSE:RETURN
2680 GOSUB780:A=USR1(0)
2690 LOCATE3,11:PRINT"*** Atenção! Arqui
vo já existe. ***":LOCATE 3,13:PRINT"Os
dados anteriores serão perdidos.":BEEP:L
OCATE3,16:PRINT"Confirma gravação (S/N)
?";
2700 A$=INKEY$
2710 IFA$="S"THENBEEP:GOSUB2230:RETURN
2720 IFA$="N"THENBEEP:RETURN
2730 GOTO2700
2740
2750 CLS:LOCATE10,1:PRINT"*** Ler arquiv
o ***":MG$="arquivo ":GOSUB2880
2760 IF A$="D"THENNO$=".DRW"ELSENO$=""
2770 GOSUB2970:XL=0
2780 IF P$<>"A:"THENMG$="leitura":GOSUB2
210:GOTO2800
2790 GOSUB2220
2800 OPEN N$ FORINPUTAS#1
2810 INPUT#1,GX,GY,C,K
2820 FORI=0TOK
2830 INPUT#1,X$:T=INSTR(X$,"*")
2840 IF T>0THEN MID$(X$,T,1)=","
2850 D$(I)=X$:NEXT:CLOSE
2860 FLAG=0:X1=GX:Y1=GY:X=128:Y=96:NS=1:
FC=0:GRID=0:CS=15:GOSUB3370:RETURN
2870 '
2880 LOCATE5,22:PRINT"<ESC> Retorna ao m
enu principal":LOCATE5,8:PRINT"(D)isco o
```

```
u (F)ita: ";
2890 A$=INPUT$(1):BEEP
2900 IFA$=CHR$(13)THENA$="D"
2910 IFA$="D"THENTN=8:P$="A:":PRINT"Disc
o":RETURN
2920 IFA$="F"THENTN=6:P$="CAS:":PRINT"Fi
ta":RETURN
2930 IFA$=CHR$(27)THEN RETURN1780 ELSE G
OTO2890
2940 '
2950 CLS:LOCATE5,0:PRINT"***  Diretório
do disco ***":LOCATE0,3:FILES"A:":PRINT:
PRINT:PRINT"Bytes livres : ";DSKF(0)*102
4:PRINT:PRINT:GOSUB790:RETURN
2960 '
2970 LOCATE5,22:PRINT"<ESC> Retorna ao m
enu principal"
2980 LOCATE5,10:PRINT"Nome do ";MG$;" :
";STRING$(TN,95);NO$
2990 N$=""
3000 A$=INKEY$:IFA$=""THEN3000
3010 G=ASC(A$)
3020 IF (G<48 OR G>57) AND (G<65 OR G>90
) AND G<>13 AND G<>8 AND G<>27 AND G<>45
 THEN3000
3030 BEEP
3040 IFG=27 THEN RETURN1780
3050 IFG=8 AND N$=""THEN2980
3060 IFG=8 THENN$=LEFT$(N$,LEN(N$)-1):LO
CATE24,10:PRINTN$+"_";:GOTO3000
3070 IFG=13 ANDN$=""THEN3000
3080 IFG=13 THEN N$=P$+N$+NO$:RETURN
3090 IF LEN(N$)=TN THEN3000
3100 N$=N$+A$
3110 LOCATE23+LEN(N$),10:PRINTA$;
3120 GOTO3000
3130 '
3140 COLORC,1,C:SCREEN2
3150 GOSUB1730:FORI=1TO5:KEY(I)ON:NEXT:F
C=0:RETURN330
3160 '
3170 NO$="Apagar tudo":MG$="apagar":GOSU
B3260
3180 A=USR1(0)
3190 LOCATE2,10:PRINT"Confirma ? (S/N) "
;:A$=INPUT$(1):BEEP
3200 IF A$="N"THENRETURN
3210 IF A$="S"THEN ERASE Q$:DIMD$(22):FL
AG=2:X=128:Y=96:C=4:K=0:GRID=0:NS=1:CS=1
5:RETURN
3220 GOTO3190
3230 '
3240 SCREEN0,0,1:COLOR15,1,1:WIDTH40:A=U
SR2(0):KEYON:POKE&HFCAB,0:END
3250 '
3260 CLS:NO$="*** "+NO$+" ***":NL=LEN(NO
$)/2:LOCATE19-NL,1:PRINTNO$
3270 IF Q$(0)<>""THEN RETURN
3280 MG$="*** Não há nada para "+MG$+" *
**":NL=LEN(MG$)/2:LOCATE 19-NL,10:PRINTM
G$:GOSUB790:RETURN1780
3290 '
3300 OPENN$FOROUTPUTAS#1
3310 PRINT#1,"49997 ' ";STRING$(14,61)
3320 PRINT#1,"49998 ' DRAW BY DRAWER"
3330 PRINT#1,"49999 ' ";STRING$(14,61)
3340 LI!=50000!:L$=STR$(LI!):L$=RIGHT$(L
$,LEN(L$)-1)
3350 Q$=STR$(S):S$="s"+RIGHT$(Q$,LEN(Q$)
-1):V$=STR$(CO):W$="a"+RIGHT$(V$,LEN(V$)
-1):PRINT#1,L$;" DRAW ";CHR$(34);S$;W$;C
HR$(34):LI!=LI!+10
3360 FORI=0TOV:L$=STR$(LI!):L$=RIGHT$(L$
,LEN(L$)-1):PRINT#1,L$;" DRAW ";CHR$(34)
;M$(I);CHR$(34):LI!=LI!+10:NEXT:CLOSE
3370 GOSUB780:LOCATE0,11:PRINT"*** Opera
ção executada com sucesso ***":GOSUB790:
RETURN
3380 '
3390 CLS:KEYOFF:COLOR15,1,1:SCREEN2,0,0
3400 FORI=1TO10:KEYI,"":NEXT
3410 DIM D$(20),M$(9):POKE&HFCAB,1
3420 X=128:Y=96:C=4:K=0:CT=0:CS=15
3430 FLAG=2:GRID=0:NS=1:EX=0:FC=0
3440 FORI=&HD000TO&HD030:READA$:POKEI,VA
L("&H"+A$):NEXT
3450 DEFUSR0=&HD000:DEFUSR1=&H156:DEFUSR
2=&H3E
3460 FORI=1TO8:READA$:B$=B$+CHR$(VAL("&H
"+A$)):NEXT:SPRITE$(0)=B$
3470 B$="":FORI=1TO8:READA$:B$=B$+CHR$(V
AL("&H"+A$)):NEXT:SPRITE$(1)=B$
3480 GOSUB3740:FORI=1TO5:KEY(I)ON:NEXT
3490 COLORC,1,C:CLS:DRAW"s4a0c4":RETURN
3500 '
3510 IFERR=5ORERR=56THENE$="Sem Disk-Bas
ic.":GOTO3610
3520 IFERR=19ORERR=69THENE$="Erro de E/S
":CLOSE:GOTO3610
3530 IFERR=53ANDXL=1THENXL=0:RESUME2670
3540 IFERR=53THENE$="Arquivo inexistente
.":GOTO3610
3550 IFERR=66THENE$="Disco cheio.":GOTO3
610
3560 IFERR=67THENE$="Diretório cheio.":G
OTO3610
3570 IFERR=68THENE$="Disco protegido.":G
OTO3610
3580 IFERR=70THENE$="Verifique disco.":G
OTO3610
```

```
3590 IFERR=99THENE$="Muito complexo.":GO
TO3610
3600 SCREEN0:COLOR15,4:E$="Erro de Proce
ssamento"
3610 CLS:LOCATE8,1:PRINT"***  A t e n ç
ã o  ***":X$="":BEEP
3620 E$="*** "+E$+" ***":NL=LEN(E$)/2
3630 LOCATE19-NL,11:PRINTE$:LOCATE5,22:P
RINT"Tecle <RETURN> para continuar."
3640 A=USR1(0)
3650 A$=INKEY$
3660 IFA$<>CHR$(13)THEN 3650
3670 BEEP:RESUME 1780
3680 '
3690 FORI=1TO3:KEY(I)OFF:NEXT
3700 IF EX=0THENCOLOR,,C:FORI=1TO5:KEY(I
)ON:NEXT:RETURN
3710 COLOR,,15:BEEP
3720 FORI=1TO20:NEXT:COLOR,,1:GOTO3690
3730 '
3740 COLOR15,1,1:SCREEN2,,0:OPEN"GRP:"FO
ROUTPUTAS#1
3750 B$="u5r12frfd3glgnl12br10"
3760 X$="u5r14fdgl14r12f3br10"
3770 A$="e2nr11e3r5f4br10nu5r2ur2ur2ur2d
r2dr2dr2u5br10nr15d3nr10d3r15br10"
3780 COLOR4:DRAW"c4s6a0":FORI=1TO4:PSET(
30+I,68+I):DRAW B$+X$+A$+X$:NEXT:DRAW"bm
31,69c1"+B$+X$+A$+X$
3790 COLOR8:FORI=1TO2:PSET(112+I,45):PRI
NT#1,"M S X":NEXT:COLOR2:A$="Sistema Edi
tor":FORI=1TOLEN(A$):PRESET(95+(I-1)*6,8
0):PRINT#1,MID$(A$,I,1):NEXT
3800 COLOR8:PSET(25,183):PRINT#1,"Tecle
<ESPAÇO> para começar"
3810 IF NOTSTRIG(0)THEN 3810
3820 CLOSE:BEEP:CLS:RETURN
3830 DATA 21,00,00,06,08,0E,0E,11
3840 DATA 00,20,E5,3E,00,CD,CD,07
3850 DATA 19,79,CD,CD,07,10,05,CD
3860 DATA 2C,D0,06,08,E1,23,7D,FE
3870 DATA FF,20,03,CD,2C,D0,7C,FE
3880 DATA 18,20,DF,C9,79,EE,01,4F,C9
3890 DATA FE,FC,F8,FC,FE,DF,8E,04
3900 DATA FE,B4,88,B4,A2,D1,BA,04
```

SCREEN IV

PROJETO

SÉRGIO DURIC CALHEIROS

Após ter preparado um ambiente de trabalho, digamos, mais esperto, passaremos a contar, a partir de agora, com novos comandos incorporados ao BASIC do SCREEN IV. Estes novos comandos permitem explorar um pouco mais a fundo os recursos que o SCREEN IV oferece em seu modo de tela exclusivo.

Como já devem saber, em face da divulgação prévia de toda a programação do projeto, os novos comandos a serem implementados nesta parte serão: os CLS ON e OFF conjugados ao WIDTH, o comando OVER e, em fim, o comando FILL. A desativação automática do SCREEN IV também será implementado, conjugado ao comando CALL SYSTEM. Além disso, o CLS normal passará a funcionar como deve, retirando a obrigação do usuário de usar o comando PRINT CHR$ (12) como seu substituto. Nada impede, porém, que o PRINT seja usado no lugar do CLS, não sendo necessária, portanto, a substituição de PRINT's já existentes pelo CLS.

No época da adaptação do WIDTH para comportar as 64 colunas, comentamos que este comando seria complementado mais tarde com os comandos CLS ON e OFF. Com estes três comandos agindo em conjunto, ganharemos um controle maior sobre a tela do computador e facilidades na criação de janelas e/ou menus, aproveitando ainda melhor, a junção gráfico/texto.

Normalmente, quando utilizamos o comando WIDTH para alterar o número de colunas da tela, um CLS é executado automaticamente, o que pode ser indesejável em alguns situações. A menos que o valor parâmetro do comando seja igual ao número de colunas atual, o CLS não será executado, quando, na realidade, nada é realmente executado.

Com os comandos CLS ON e OFF, podemos habilitar e desabilitar respectivamente a execução automática do CLS, após uma mudança do número de colunas da tela através do WIDTH. Com isso, a criação de uma janela, ou melhor, de uma área de tela mais restrita, passa a ser mais automática e mais controlado.

Por outro lado, o comando OVER completam outra característica do programa. No SCREEN IV, assim como nas telas de texto, quando um caracter é impresso numa posição da tela, o padrão do caracter antigo desaparece, como se o micro limpasse aquela posição e, em seguida, aparece o padrão correspondente ao novo caracter. Devido à possibilidade da utilização simultânea de gráficos e texto, torna-se necessário uma instrução que permita a impressão de caracteres sem que haja apagamento do padrão antigo, preservando o shape existente naquela posição. Com o comando OVER, pode-se inibir ou habilitar o apagamento total ou parcial do padrão antigo, seja qual for. A sobreposição do caracter é controlado pelo parâmetro passado ao comando. Para sobrepor totalmente o padrão existente, ou seja, sem limpeza, digite OVER 15 e, para realizar uma impressão com limpeza total da área do caracter, use o comando OVER0 que é o modo normal de operação.

Um uso avançado deste comando, requer um mínimo de conhecimento em manipulação de números binários por parte do leitor. A faixa de variação do parâmetro (ou arranjo) do comando OVER está entre 0 e 15. Para compor estes números, são necessários pelo menos 4 dígitos binários, ou bits, que é o número exato de pontos necessários à composição de um caracter em 64 colunas. Um caracter nesta forma ocupa uma matriz de 4 x 8 pontos. Desta forma, sabendo controlar esses bits, teremos condições de controlar qual parte do padrão anterior vai ser preservado e qual não. A exemplo do OVER0 do OVER 15, observe que, em binário, são representados por 0000B e 1111B respectivamente. Com isso, podemos perceber que cada 0 representa uma coluna de matriz a ser apagada antes de imprimir o padrão do caracter pertencente àquela coluna. O mesmo acontece com cada 1 do número, que indica que coluna será preservada.

Como exemplo adicional, experimente o comando OVER 9. O arranjo convertido para binário, nos dá o número 1001B. Com isso, observamos que apenas o interior do caracter será apagada para dar ao novo que está para ser impresso. Observe ainda que, o parâmetro do comando OVER é um número qualquer, como acontece com o comando COLOR por exemplo. Portanto, a visualização do formato do número fica a cargo do programador, bem como a escolha de como passará o dado ao comando. Para o BASIC, o comando OVER 15 tem o mesmíssimo efeito do comando OVER &B1111, ou OVER &H0F.

Como advertência, evite utilizar o comando OVER no modo direto. Caso o comando seja utilizado com parâmetro diferente de zero, a tela ficará borrado pelo cursor, tornando a leitura do que for digitado difícil ou mesmo impossível. Para normalizar a situação, basta digitar OVER 0. Mesmo que não seja possível ler o que está escrito na tela, o micro ainda continuará a reconhecer os comandos dados.

Finalmente, temos o comando FILL, que é um comando semelhante ao comando PAINT já existente no BASIC. Sua sintaxe é FILL <STEP>(X,Y)<,COR>, praticamente a mesma do PAINT. Sua principal vantagem em relação ao PAINT é o ganho de velocidade, que é cerca de oito vezes superior. Outra característica, é a possibilidade de encher uma área da tela com uma determinada cor, sem se preocupar com a cor dos limites físicos. Com isso, a construção de telas gráficas passa a ser bem mais fácil e rápido, pois não há preocupação com a borda.

Naturalmente, há uma restrição, ou melhor, um fato que justifica o ganho de velocidade. Este ganho é conseguido porque o comando não perde tempo em verificar se a área que ele está preenchendo possui algum "buraco" por onde pode haver um escape. O comando apenas realiza a procura de espaço na vertical, procurando expandir-se lateralmente até encontrar os limites. Assim que atingir o topo e o chão da figura, o comando termina. Este "topo" ou "chão", pode ser um único ponto que esteja na linha de expansão do comando. Com isso, antes de utilizar o comando FILL, deve-se estudar os melhores pontos para localizar seu ponto de referência. Para conseguir o preenchimento total de uma figura irregular, poderá ser necessário o uso de dois ou mais comandos FILL corretamente localizados. Mesmo que dê um pouco mais de trabalho ao programador, na maioria das vezes será largamente compensador, tanto em termos de velocidade quanto eficiência. Para compreender melhor como localizar bem o ponto do FILL, estude os programas dos listagens BASIC.

A digitação dos blocos deve seguir a maneira a que o leitor já esteja acostumado. Terminando a digitação, leia a descrição dos comandos até certificar-se que conhece a sintaxe, a função e principalmente as limitações dos comandos novos. Isto é importante particularmente no caso dos comandos FILL e OVER, que requerem uma boa prática por parte do programador. Com isso, execute o programa, digite os programas exemplo e comece a explorar os novos comandos como melhor lhe convier.

Este novo conjunto de comandos somente estará disponível no ambiente gerado pelo SCREEN IV. Tudo que for digitado, poderá ser visto através da listagem do programa em outros modos de tela. Se o programa for executado em outras telas que não seja a do SCREEN IV, estes novos comandos não serão reconhecidos.

Vale ainda lembrar que esta é a antepenúltima parte do projeto SCREEN IV. Após esta, restam as partes de recuperação de telas em fita cassete e da impressão de telas gráficos. Apesar de todo o projeto já estar totalmente pronto desde o início, não quer dizer que seja definitivo. O sistema oferece a possibilidade de inclusão de novos comandos além daqueles já implementados. Os leitores que possuirem alguma rotina ou comando que possa ser implementado ao BASIC através do SCREEN IV, e desejarem divulgá-las, escrevam para que possamos examiná-los. Daremos prioridade para os programas que julgarmos serem mais úteis aos demais leitores. Até a próxima.

PROGRAMA 1

```
10 REM Testa comando OVER
20 CLS:KEY OFF:WIDTH 64:OVER Ø
30 LINE (Ø,5)-(255,5)
40 PRINT @ 10," Comando OVER Ø "
50 OVER 15
60 PRINT @ 40," Comando OVER 15 "
70 OVER Ø
80 PRINT @ 64,"Pressione
   uma tecla..."
90 A$=INPUT$(1)
```

PROGRAMA 2

```
10 REM Testa comando FILL
20 CLS:KEY OFF:COLOR 15
30 CIRCLE (127,95),90
40 FILL (127,95),2
50 FILL (255,95),3
60 FILL (Ø,95),3
70 PRINT @ Ø,"Pressione
   uma tecla..."
80 A$=INPUT$(1)
```

PROGRAMA3

```
10 REM Estabelece parada do FILL
20 REM a partir do ponto de localizacao.
30 REM
40 REM Comando MAL localizado:
50 KEY OFF:WIDTH 64
60 LINE (Ø,Ø)-(255,191)
70 FILL (127,Ø)
80 PRINT @ Ø,"Note o ponto de
    parada (127,95).
90 PRITN @ 64,"Pressione
     uma tecla..."
100 A$=INPUT$(1)
110 REM
120 REM Comando BEM localizado:
130 CLS
140 LINE (Ø,Ø)-(255,191)
150 FILL (255,Ø)
160 PRINT @ Ø,"O ponto agora é
     (255,191).
170 PRINT @ 64,"Pressione
     uma tecla...
180 A$=INPUT$(1)
190 CLS
200 REM Exemplo explicito de parada
     do comando
210 PSET (127,95)
220 FILL (127,140),2
230  PRINT  @ Ø,"Qualquer ponto na
     linha de expansão serve como
     critério de parada!
```

BLOCO 1

```
412Ø 11 C3 38 11 C3 27 14 C3
4128 88 14 C3 67 15 C3 ØØ ØØ
```

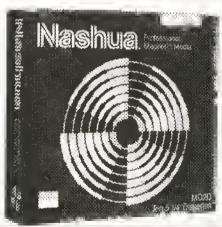
BLOCO 2

```
417Ø FD 88 FD C2 FD D8 FD E5
4178 FD 7Ø FF 89 FF 43 FF ØØ
```

BLOCO 3

```
5567 22 7F 38 11 AØ Ø2 D5 FE
556F 9F CA DA 15 FE 46 CA 45
5577 16 FE 4F CA 91 15 FE CA
557F CA Ø6 16 FE 5F CA Ø6 16
5587 ØØ ØØ ØØ ØØ ØØ D1 C9 56
558F 45 52 F5 11 8E 15 23 Ø6
5597 Ø3 1A BE C2 23 16 23 13
559F 1Ø F7 2B DD 21 55 47 CD
55A7 7B Ø2 22 7F 38 21 ØF ØØ
55AF CD 84 Ø2 3Ø Ø7 DD 21 67
55B7 4Ø CD 7B Ø2 AF F6 FØ B3
55BF 32 6A 38 CB 23 CB 23 CB
55C7 23 CB 23 AF F6 ØF B3 32
55CF 68 38 F1 DD 21 82 46 2A
55D7 7F 38 C9 3A A9 FD FE F7
55DF 3E 9F 2Ø 1B F7 ØØ 66 46
55E7 FE 95 28 43 FE EB 28 46
55EF CD C6 Ø2 2A 7F 38 23 22
55F7 7F 38 DD 21 82 46 C9 D1
55FF C9 53 59 53 54 45 4D F5
56Ø7 DD 21 66 46 CD 7B Ø2 11
56ØF ØØ 16 Ø6 Ø6 1A 8E 2Ø ØC
5617 13 23 1Ø F8 CD 4E Ø2 21
561F CØ FF 36 C9 2A 7F 38 F1
5627 D1 ØØ ØØ ØØ ØØ ØØ C9 AF
562F 3D 32 4D 38 18 Ø4 AF 32
5637 4D 38 23 22 7F 38 DD 21
563F 82 46 C9 49 4C 4C F5 11
5647 42 16 23 Ø6 Ø3 1A 8E 2Ø
564F D3 23 13 1Ø F8 2B DD 21
5657 66 46 CD 78 Ø2 DD 21 9C
565F 57 CD 7B Ø2 DD 21 67 4Ø
5667 E5 C5 43 ED 43 6D 38 21
566F 8F ØØ CD 84 Ø2 DA C7 16
5677 D1 21 FF ØØ CD 84 Ø2 DA
567F C7 16 E1 2B DD 21 66 46
5687 CD 7B Ø2 28 26 FE 2C DD
568F 21 55 4Ø C2 C7 16 DD 21
5697 55 47 CD 7B Ø2 22 7F 38
569F 21 ØF ØØ CD 84 Ø2 7B 2A
56A7 7F 38 3Ø ØA DD 21 5A 47
56AF C3 C7 16 3A E9 F3 32 6F
56B7 38 22 7F 38 CD CE 16 F1
56BF DD 21 82 46 2A 7F 38 C9
56C7 CD 78 Ø2 2A 7F 38 C9 3A
56CF AF FC FE Ø2 28 ØA 2A 7F
56D7 38 DD 21 5A 47 C3 7B Ø2
56DF CD 2Ø 19 CD 7Ø 18 CD 18
56E7 17 2A 7Ø 38 7D E6 Ø7 B4
56EF 28 Ø8 CD FB 17 CD A5 18
56F7 28 EC CD 7D 18 3A 6E 38
56FF FE 8F C8 CD 1Ø 18 CD A5
57Ø7 18 CØ CD 18 17 2A 7Ø 38
57ØF 7D E6 Ø7 84 FE 1E 2Ø EB
5717 C9 CD E1 17 CD 9D 18 2Ø
571F ØC CD 22 18 CD E1 17 CD
5727 D7 17 C3 46 18 CD 66 17
572F C2 4B 17 3A 7D 38 A1 79
5737 28 Ø9 CD 94 17 CD C8 17
573F C3 DF 18 CD 75 17 CD D7
5747 17 C3 46 18 37 CB 1Ø CB
574F 19 3Ø F9 BØ 4F Ø6 ØØ 37
5757 CB 18 C8 11 3Ø F9 BØ 2F
575F 83 CD 81 19 C3 C7 18 4F
5767 3A 7E 38 A1 C8 3A 7D 38
576F A1 Ø6 ØØ 79 59 C9 Ø6 ØØ
5777 4F FE FF C8 CA 8A 18 37
577F CB 1Ø C8 19 3Ø F9 CB 38
5787 4F BØ FE FF CA Ø9 19 CD
578F 81 19 C3 C7 18 Ø6 ØØ 4F
5797 FE FF C8 FE Ø1 CA 8A 18
579F 37 C8 18 CB 11 3Ø F9 CB
57A7 2Ø 4F BØ FE FF CA Ø9 19
57AF CD 81 19 C3 C7 18 22 7Ø
57B7 38 11 ØØ 2Ø 19 22 72 38
57BF C9 22 74 38 11 ØØ 2Ø 19
57C7 22 76 38 C9 A7 2A 74 38
57CF 11 Ø8 ØØ ED 52 C3 CØ 17
57D7 2A 74 38 11 Ø8 ØØ 19 C3
57DF CØ 17 2A 72 38 22 76 38
57E7 2A 7Ø 38 22 74 38 C9 2A
57EF 76 38 22 72 38 2A 74 38
57F7 22 7Ø 38 C9 2A 7Ø 38 2B
57FF 7D E6 Ø7 FE Ø7 C2 B5 17
58Ø7 11 F8 ØØ A7 ED 52 C3 85
58ØF 17 2A 7Ø 38 23 7D E6 Ø7
5817 A7 C2 85 17 11 F8 ØØ 19
581F C3 B5 17 2A 74 38 7D E6
5827 F8 28 12 CD 9D 18 C2 75
582F 17 3A 6F 38 4F CD 8A 18
5837 CD CB 17 18 E6 CD 9D 18
583F C2 75 17 79 C3 8A 18 2A
5847 74 38 7D E6 F8 FE F8 28
584F 16 7D E6 F8 C8 CD 9D 18
```

```
5857 C2 94 17 3A 6F 38 4F CD
585F 8A 18 CD D7 17 18 EØ CD
5867 9D 18 C2 94 17 79 C3 8A
586F 18 2A 7Ø 38 22 78 38 2A
5877 72 38 22 7A 38 C9 2A 78
587F 38 22 7Ø 38 2A 7A 38 22
5887 72 38 C9 CD C7 18 2A 74
588F 38 3E FF C3 81 19 2A 7Ø
5897 38 CD 7B 19 A7 C9 2A 74
589F 38 CD 7B 19 A7 C9 2A 7Ø
58A7 38 3A 7C 38 4F CD 7B 19
58AF A1 C9 2A 76 38 CD 7B 19
58B7 CB 27 C8 27 CB 27 CB 27
58BF 4F 3A 6F 38 B1 C3 81 19
58C7 2A 76 38 CD 7B 19 E6 ØF
58CF 47 3A 6F 38 CB 27 CB 27
58D7 C8 27 CB 27 BØ C3 81 19
58DF 2A 74 38 7D E6 F8 28 18
58E7 7D E6 F8 FE F8 C8 CD 9D
58EF 18 C2 75 17 3A 6F 38 4F
58F7 CD 8A 18 CD C8 17 18 EØ
58FF CD 9D 18 C2 75 17 79 C3
59Ø7 8A 18 2A 76 38 CD 7B 19
59ØF E6 FØ 4F 3A 6F 38 81 C3
5917 81 19 ØØ CD 81 19 C3 B1
591F 18 A7 3A 6E 38 11 F8 ØØ
5927 21 ØØ ØØ 47 Ø4 7D A7 28
592F Ø6 E6 Ø7 A7 2Ø Ø1 19 23
5937 1Ø F3 3A 6D 38 11 Ø8 ØØ
593F 28 C8 3F CB 3F CB 3F 3C
5947 A7 47 ED 52 19 1Ø FD 3A
594F 6D 38 E6 Ø7 CD 85 17 3C
5957 47 11 FF ØØ ØE Ø1 CB Ø9
595F 37 CB 1A A7 CB 1B 1Ø F6
5967 CB 22 ED 53 7D 38 79 32
596F 7C 38 2A 7Ø 38 CD 7B 19
5977 A1 C8 D1 C9 CD 87 19 DB
597F 98 C9 CD 92 19 D3 98 C9
5987 7D F3 D3 99 7C E6 3F D3
598F 99 FB C9 F5 F3 7D D3 99
5997 7C E6 3F F6 4Ø D3 99 F1
599F F8 C9 ØØ ØØ ØØ ØØ ØØ ØØ

Soma total:Ø1CE31
```

CLUBE DO LEITOR
O CARTÃO DO MSX

MANÍACOS DO MSX
15% desconto na compra de jogos.
20% desconto na compra de jogos especiais.
10% desconto na compra de programas de autores nacionais.
15% desconto na compra de aplicativos.

CONECTOR IND. E COM. LTDA
5% desconto na compra de kit de drive para MSX.

CIBERTRON ELETRÔNICA LTDA
15% desconto na compra de software.

YOUNGSOFT
30% desconto nas compras de software.
10% desconto na inscrição no clube de usuários.

NEMESIS INFORMÁTICA LTDA
10% desconto em seus produtos.

RECURSOS DIGITAIS
5% desconto na compra de periféricos.
10% desconto na compra de softs de outras empresas.
30% desconto na compra de softs da Redi Universoft.

TACTO INFORMÁTICA COM. LTDA.
10% desconto na compra de qualquer produto ou curso.

PAULISOFT INFORMÁTICA LTDA
10% desconto na compra de software, exceto promoção.

DISCOVERY INFORMÁTICA
10% desconto em seus produtos.

EDITORA ALEPH
15% desconto em suas publicações.

REVOLUTION
20% desconto nas compras de software.

NEWSOFT
10% desconto na compra dos jogos comuns.
20% desconto nos jogos especiais.
25% desconto nos aplicativos.
30% desconto na compra de livros.
5% desconto na compra de periféricos e suprimentos.

THUNDERSOFT
20% desconto em todos os seus produtos.

NEWDATA
5% desconto nos produtos de representação/revenda.
10% desconto nos seus produtos.

ESPACIAL ELETRÔNICA
20% desconto nos seus produtos.

INFORTELLES
15% desconto em geral.

GAME OF TIME
10% desconto em geral.

SOFTMARK
12% desconto nos seus produtos.

SOFT DESIGNS
15% desconto na compra de software e serviços.

MSX INFORMÁTICA
10% desconto em hardware.
20% desconto em software da MSX INFORMÁTICA e ou
10% desconto em software de outras EMPRESAS.
10% desconto em assistência técnica e suprimentos.

Se você ainda não tem um cartão,
faça logo a sua assinatura de CPU e receba o seu!

MSX DEBUG

PROJETO

SÉRGIO DURIC CALHEIROS

Passada a avalanche de BYTES quer derrubamos sobre o leitor na última parte do projeto, agora nos ocuparemos em implementar um nova e (bem) simples comando ao MSXDEBUG. Este comando é o comando APPEND. Como o nome sugere, este comando se encarrega de APPENDar, ou melhor, de gravar um bloco de dados da memória no final de um arquivo já existente no disco.

Mas afinal, poderiam perguntar, que utilidade teria um comando que junta um bloco de memória com um arquivo? Talvez sua utilidade não seja óbvia no momento, mas asseguramos que a hora de usar o APPEND surgirá mais cedo ou mais tarde.

O sistema operacional (DOS) do MSX, organiza os dados exatamente da mesma maneira que o sistema operacional dos micros compatíveis com o IBM PC, o que possibilita que os arquivos de um possam ser manipulados pelo outro. Isto é explicado pelo fato da organização lógica de cada um ser a mesma, o que quer dizer que o MSX herdou, além das qualidades, suas deficiências.

A quantidade total de informação que um disquete comum pode reter é bem maior que aquela que é oferecida para guardar os arquivos do usuário. Isto ocorre, porque parte do espaço total é reservado para dados fixos do disco, destinado para identificar parte físicas como um setor ou uma trilha e, ainda, para identificar as partes lógicas como clusters e arquivos.

Além de todo esse espaço, digamos, roubado do usuário, ainda existe um detalhe que poucos conhecem. Quando um arquivo novo é incluído no disco, há a ocupação de tantas clusters quantas forem necessárias para armazená-lo. Do mesmo maneira que um arquivo tem um tamanho fixo, o cluster, que é uma unidade lógica, também tem. Isto quer dizer que, mesmo que o arquivo contenha apenas um único BYTE, o sistema operacional sempre reservará o espaço mínimo de um cluster para armazená-lo. Nos discos de dupla face, cada cluster contém dois setores de 512 bytees, ou seja, 1 Kbyte para cada cluster no total.

Mas o desperdício de espaço não para aí. O própria usuário contribui para diminuir este espaço tão precioso por simples falta de controle ou mesmo de informação ou de técnicas para aproveitar ao máximo o espaço disponível. Já que não é possível mudar o que já está estabelecido pelo própria hardware, podemos ao menos mudar o comportamento do usuário no momento da criação e armazenamento dos seus arquivos em disco.

Tomemos um exemplo numérico, a fim de ilustrar este processo e de como é possível economizar clusters sem ter que eliminar nossos programas. Voltando ao espaço oferecido ao usuário, vejamos quanto sobra no total. Um disco de face dupla contém 40 trilhas por face que, por sua vez, contém 9 setores por trilha, o que nos dá os 360 Kbytes já conhecidos. Descontado o espaço destinado ao boot, fat e diretório, além da cópia da fat e da própria DOS, nos sobram, digamos, uns 350 Kbytes.

Se dividirmos este espaço entre arquivos de 16 Kbytes, que é a média de um arquivo para o MSX, teríamos, teoricamente, cerca de 22 arquivos por disco. Entretanto, na prática, não é isto que acontece, pois parece que cada arquivo ocupa mais do que deveria ocupar. Isso se deve ao fato de que a maioria desses arquivos são arquivos gerados pelo BASIC, principalmente se foram jogos. Como vimos em partes anteriores, o BASIC precisa de uma parte do arquivo, justamente para saber se o tal arquivo foi mesmo gerado por ele, além dos endereços de localização do bloco. Observem que um arquivo de 16 Kbytes ocupa exatamente 16 clusters. Mas, se o tal arquivo foi realmente gerado pelo BASIC, serão necessários 17 clusters, um a mais para guardar o acréscimo de 7 BYTES feito pelo BASIC, perceberam?

Observem ainda que, um jogo razoável contém uns 32 Kbytes, que normalmente são divididos em dois blocos de 16 Kbytes cada. Além dos blocos em binário, também é necessário um pequeno programa em BASIC para carregar os dois primeiros. Com isso, há um desperdício de pelo menos 3 clusters para cada jogo. Um para cada bloco binário e, no mínimo, mais outro para o programa BASIC, se não for grande.

Se pudéssemos juntar vários blocos num só, estaríamos, teoricamente, economizando um cluster para cada bloco. Ainda, se pudéssemos dispensar o programa BASIC de carregamento, estaríamos economizando mais outro cluster. Com isso, o trabalho de carregamento ficaria a cargo do DOS, o que daria um ar mais profissional ao micro, sem ter que passar pelo BASIC e sem necessidade de quebrar o programa em várias partes.

Desta maneira, com uma média de 10 jogos completos por disco, acabaríamos economizando cerca de 20 clusters, o que possibilitaria a inclusão de um novo arquivo de graça. Além disso, os discos passariam a ficar mais organizados, pois poderíamos guardar mais informação e com menos arquivos aparecendo cada vez que listarmos um diretório. Muitas vezes, na prática, nem sempre poderemos economizar tantos clusters quantos forem os blocos, porque nem sempre será possível eliminar uma quantidade de dados suficiente para tal.

A necessidade da divisão dos programas surgiu, porque o BASIC do MSX não comporta um bloco de memória maior que 28 Kbytes de uma vez só. Apesar do micro possuir 64 Kbytes de memória RAM, os primeiros 32 Kbytes estão em ROM, lugar onde está o interpretador BASIC. Outro fator que influiu nesta divisão, foi o não disponibilidade de drive para o MSX, o que limitou, ou melhor, obrigou o programador a trabalhar no ambiente do BASIC. Um exemplo clássico deste processo, foi a adaptação dos jogos contidos em cartucho para serem carregados a partir da cassete.

Com o advento dos drives, e conseqüentemente, do DOS, tornou-se possível explorar toda RAM de forma mais natural. Um aspecto do DOS é que, quando está ativo, toda memória de 64 Kbytes está habilitado, onde os programas com extensão COM são carregados diretamente, praticamente independentes do seu tamanho.

Todo programa COM é carregado e executado a partir do endereço 100H. Geralmente, os programas em cartucho carregados pelo BASIC rodam na página 1, cujo início está no endereço 4000H. Para fazer com que os programas carregados pelo DOS sejam posicionados corretamente em seus respectivos lugares, será necessário juntar ao bloco todo uma pequena rotina que faça a transferência e habilite ou não a BIOS antes de sua execução. É por este motivo que nem sempre é possível economizar tantos clusters quantos forem os blocos. Talvez não seja vantagem converter programas que contenham apenas um bloco sob o ponto de vista de economia, mas certamente compensará em termos de organização e pelo fato de nos livrarmos do BASIC. Além é claro, do carregamento ser mais fácil, prático e rápido.

O trabalho de união dos blocos de um jogo nem sempre é tarefa simples. Às vezes pode parecer impossível realizar o trabalho. Entretanto, esteja certo que, por mais dividido que esteja um programa, sempre será possível fazê-lo. O comando APPEND será o primeiro comando de uma série de comandos que serão implementados ao MSXDEBUG para facilitar a vida do usuário nesta tarefa.

Por enquanto, não adianta usar o comando APPEND, pois muito mais é necessário para converter um programa para o tipo COM. Na realidade, o comando APPEND será utilizado apenas em casos extremos, ou seja, naqueles casos em que mesmo no MSXDEBUG não houver espaço memória suficiente para unir todo o programa de uma só vez. Quando isso ocorrer,

o que é raro, teremos que recorrer ao espaço livre do disco.

Antes de mostrarmos como utilizá-lo para gerar trabalho útil, teremos que nos familiarizar com as técnicas utilizadas para divisão dos arquivos, para depois uni-los, sabendo distinguir que parte deve ser considerada como pertencente ao programa e que parte não. Isto é particularmente importante porque, sabendo descartar dados e código inúteis, estaremos diminuindo o espaço ocupado pelo programa.

A implementação do comando deve ser imediata. Observem que este comando também é relocável, mas procure colocá-lo em seu local original, mostrado no bloco 1. Somente para fins de controle sobre nosso trabalho, o BYTE marcador do fim da tabela de comandos deverá ter se deslocado para o endereço 4D1DH após esta implementação. A entrada do comando fica no endereço 17C0H devendo ser definida na tabela das chamadas que deve estar em 4C9CH. Naturalmente, estes endereços deverão coincidir somente com o dos leitores que estão seguindo o projeto à risca. Após a digitação verifique a SOMA e aproveite para mudar a versão do programa para 1.6. Não creio que este comando deve ser colocado na tabela das teclas de função, justamente pela freqüência de uso.

O funcionamento pode ser verificado com um simples teste. Primeiramente, preencha a página 1 com o BYTE 55H. Salve uma parte desta área com o comando DSAVE. A seguir, preencha esta mesma área com outro valor, digamos 0AAH. Feito isso, use o comando APPEND para incluir esta área no fim do arquivo previamente criado. Sua sintaxe é a mesma do comando DSAVE. É importante saber que o comando não cria arquivos no disco, mas apenas junta blocos em arquivos já existentes.

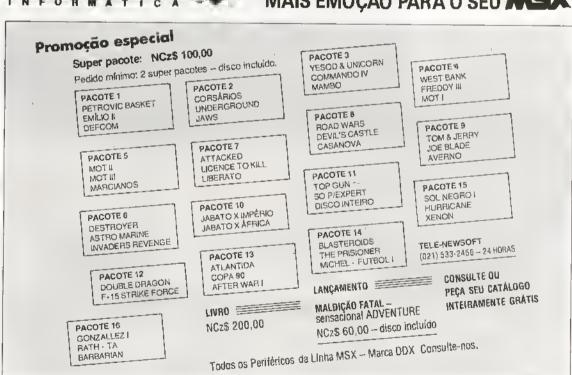
No próximo número implementaremos o comando SAVECOM encarregado de salvar os blocos já unidos de uma só vez, já com a extensão COM e junto com o programa de transferência. Além disso, veremos o processo de união dos blocos e identificação das áreas importantes. Esteja certo que entendeu a filosofia da conversão de arquivos. Caso tenha alguma dúvida, não hesite em escrever. Até lá.

SÉRGIO DURIC CALHEIROS

```
Bloco 1

47C0 CD A5 08 CD 66 07 CD 72
47C8 07 CD B6 06 CD 66 07 CD
47D0 4A 07 CD 3F 06 11 45 0C
47D8 C2 06 0B 11 80 00 0E 1A
47E0 CD 05 00 11 5C 00 CD 48
47E8 06 28 F8 C3 89 06 00 00

Soma total:000FA7
```

JOGO

CASTLE II

(1ª Parte)

CARLOS DOS SANTOS

Entre a grande variedade de tipos de jogos para o MSX, os que mais nos atraem são aqueles que exigem vários tipos de raciocínio. Dentre estes tipos de jogos, destacaríamos como um dos mais completos, o jogo CASTLE (fig. 1). Ora vejamos: ele tem um bom visual, em determinados momentos (não é constante) exige precisão de movimentos, tem muitos obstáculos (100 salas), muitas surpresas (lógicas de movimento variado) e, principalmente, muito desafio de lógica e, obviamente, de raciocínio.

As duas versões do jogo CASTLE (I e II) são muito boas, sendo que o primeiro é bem mais simples. Tão simples que necessitamos de quatro dias (somando os tempos que nos dedicamos a ele), para terminá-lo e, para o segundo, levamos três semanas. Que vergonha! ?

Este artigo é dedicado a você, que, "por acaso", teve "alguma" dificuldade para ultrapassar "uma" ou "outra" das 100 salas do jogo CASTLE II, mas não se iluda ou pense que vamos tirar a graça da sua diversão. Mesmo com os comentários que faremos sobre que caminhos seguir e que objetivos conquistar, este jogo exigirá de você muita destreza de movimentos, raciocínio e paciência. Porém, se você chegar a uma sala e não conseguir ultrapassá-la após mais de 10 tentativas ou mais de quatro horas, pelo menos, escreva imediatamente para a redação desta revista solicitando a forma de ultrapassar aquela sala e, se der tempo, no próximo número, você terá a resposta e poderá acompanhar as dicas do próximo artigo.

FIGURA 1 — APRESENTAÇÃO

REGRAS BÁSICAS

Antes de mais nada, uma breve síntese do objetivo geral do jogo, auxiliado pelo desenho da figura 2, que mostra os tipos de personagens, prêmios, obstáculos e outros, que aparecerão no interior das salas.

OBJETIVOS: PRÍNCIPE — personagem principal que é guiado por você.

MAPA — permite você acompanhar as salas por onde já passou.

CHAVES — permite abrir as PORTAS de mesma cor.

PORTAS — são abertas com as CHAVES de mesma cor.

BORBOLETAS — entregam uma CHAVE laranja.

PRINCESA — o "obcessão" do PRÍNCIPE.

PRÊMIOS: BARRAS DE OURO — dão 800 pontos. ANÉIS — dão 400 pontos. CRUCIEIXOS — dão 200 pontos. POTES VERDES — dá uma vida extra. OXIGÊNIO — permite passar pela água. ARCO-ÍRIS — quando o PRÍNCIPE passa por sua cortina, nenhum inimigo o afeta.

AUXÍLIOS: TIJOLINHO, BOLO, SACO, JARRÃO, BARRIL — todos estes auxílios podem ser empurrados e/ou transportados para qualquer lugar, dependendo da sua necessidade. Com excessão do BOLO, o PRÍNCIPE pode subir em qualquer um deles.

TRANSPORTES: HORIZONTAL FLUTUANTE — o PRÍNCIPE pode andar junto, sobre ele.

VERTICAL COM PARAFUSO — o PRÍNCIPE não pode atravessar o parafuso.

VERTICAL COM HASTE — o PRÍNCIPE pode atravessar a sua haste.

ONDINHA — quando parada, o PRÍNCIPE a transpassa e o REIZINHO é destruído se estiver sobre ela. Quando ondulando, tanto o PRÍNCIPE quanto o REIZINHO, caminham por sobre ela.

ESTEIRA — transporta o que estiver sobre ela, porém, o PRÍNCIPE pode ficar parado sobre ela se for mantida pressionada a direção inversa à direção do movimento da ESTEIRA.

GUINDASTE — o contra peso (à esquerda) tem que ser empurrado para fazer a "caçamba" do GUINDASTE subir ou descer. Pode, inclusive, ser tirado deste lugar e servir como um degrau, tal como o TIJOLINHO, BARRIL, etc...

INIMIGOS: SOLDADO, RATÃO, REIZINHO, BRUXINHA — todos estes têm

FIGURA 2 — PERSONAGENS

movimento cego ou seja: não perseguem efetivamente, apenas atrapalham o caminho do PRÍNCIPE. O REIZINHO, em alguns casos, salta (um pouquinho) para tentar pegar o PRÍNCIPE, quando este passa por cima dele. FOGUINHO — esta bolinha de fogo persegue mesmo, mas se houver um obstáculo entre ela e o PRÍNCIPE, às vezes, ela esquece a perseguição.
PLANTINHA — ela dorme e acorda. Não pode ser pisada nunca. Quando está dormindo, ocupa menos espaço e facilita um salto por sobre ela. Pode ser destruída se você empurrar alguma caixa sobre ela.
PERIGOS: ESPETOS — todos eles são mortais.
Você tem que guiar o PRÍNCIPE para encontrar e libertar uma PRINCESINHA para se casarem e serem felizes e etc.. Para tal, terá de passar por todas as 100 salas do castelo (algumas por várias vezes). Muitas destas salas têm passagens livres para outras salas, mas, na maioria delas, estas passagens estão fechadas por PORTAS coloridas que, obviamente, exigem uma CHAVE de mesma cor para abri-las. Existem passagens que são só de ida e outras que são armadilhas do tipo cair do 8º andar até o 3º andar e ter que refazer todo o caminho.

Na figura 3, mostramos um mapa de todas as salas e suas interligações (passagens ou PORTAS). As letras e os números em torno do quadro vão permitir identificar cada sala para uma ou outra dica e etc... Dentro de cada sala colocamos um caracter de referência quanto ao que existe nela e estes caracteres expressam o seguinte:

* = INÍCIO
P = PRECISÃO DE MOVIMENTOS
Q = QUEBRA-CABEÇA
L = CHAVE LARANJA

As salas mais calmas estão sem referência.

Em algumas salas existem armadilhas muito perigosas e em outras existem quebra-cabeças (muito bem bolados) que, em alguns casos, por mais rápido que seja o seu raciocínio ou mesmo que contemos tudo que você tem que fazer nelo, você deverá gastar uns 20 minutos. Durante a nossa primeira tentativa, houveram salas nas quais necessitamos de uns 3 dias de tentativas até ter aquele "estalo" e conseguir "matar a charada".

Como não conhecemos ninguém que tenha terminado este jogo num único dia, aqui está um recurso muito oportuno. O jogo permite que você grave em fita cassete a situação atual e, obviamente, continuar num outro momento. Esta função pode, inclusive, trazer menos tristeza quando acontecem aquelas quedas ou faltas de luz ou se, no afã de conseguir ultrapassar uma determinada sala, você conseguir acabar com as 256 vidas que estão disponíveis no início do jogo. Sugerimos que, de tempos em tempos, grave a situação atual do jogo. Isto, no mínimo, servirá para você descansar a "cuca".

Sempre que você quiser salvar o jogo na fita, basta pressionar a tecla F4 e, para a leitura do jogo, que só pode ser feita durante a apresentação do início do jogo (fig. 1), primeiro reposicione a fita no início de uma gravação (que tem um apito contínuo e longo como característica) e tecle F3.

Muita atenção ao usar as teclas de função, pois se você teclar F2, o jogo é encerrado. Se você teclar F1, você mata o PRÍNCIPE. Como você é quem guia (encarna) o PRÍNCIPE, você se suicida no jogo. O F1 é muito importante (apesar de se perder uma vida, pois, em alguns casos, você pode entrar numa "sinuca" ou seja, fica preso num lugar que não tem como sair e, nesses casos, o jeito é suicidar-se, o que provoca a sua ressurreição (se ainda tiver vidas extras), na entrada da mesma sala. E mais um detalhe: se, por acaso, você tiver destruído alguma coisa naquela sala e chegar à conclusão que não deveria ter destruído, suicidando-se, o que foi destruído reaparece no seu lugar de origem. Mas, lembre-se, isto só é válido enquanto você não sair da mesma sala e somente se sua morte ocorrer por intermédio da tecla F1.

A tecla F5 paralisa o jogo (pausa) até qualquer outra tecla ser pressionada. Mantendo-se pressionada a tecla CTRL (control), a velocidade dos movimentos é aumentada e com as teclas CTRL

	A	B	C	D	E	F	G	H	I	J	
0	Q	Q	P		P			P	Q		0
1	P	P	P	Q	Q		P	P	P	P	1
2	P	P	Q	Q	P	Q		Q	P	P	2
3	Q	L	P	Q	Q	Q	Q	P	Q	P	3
4	Q	Q	Q	P	P	P	P	P	Q	P	4
5	Q	Q	Q	Q	Q	Q	Q	P			5
6		P	Q		Q	P	P		P	P	6
7	*			P	Q		P		P	P	7
8					Q	Q	Q	P	L	P	8
9			Q	P		P	Q	P	Q	Q	9
	A	B	C	D	E	F	G	H	I	J	

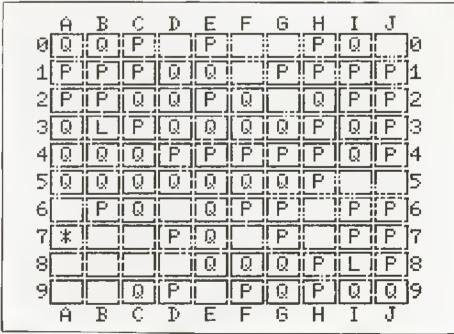



FIGURA 3 — MAPA GERAL

e SHIFT pressionados simultaneamente a velocidade aumenta mais ainda. A musiquinha do jogo silencia quando se pressiona CAPS.

Movimenta-se o PRÍNCIPE para a direita ou para a esquerda (joystick ou teclas de direção), para saltar (botão de tiro ou barra de espaço) e para baixo é só "deixar cair".

Um dos segredos do jogo está em saber usar corretamente o movimento de salto ou subida. Dependendo do tempo que você mantiver pressionada a barra de espaço ou o botão de tiro, o PRÍNCIPE alcança alturas diferentes, pode permanecer no alto por um certo limite de tempo que pode ser diminuído com o ato de soltar a tecla ou o botão de tiro, e pode ser direcionado simultaneamente durante a subida ou na descida, o que, às vezes, é fundamental para se conseguir certas posições.

O JOGO

Nesta primeira parte do artigo, vamos comentar dicas para você alcançar a sala E9 na porta de baixo sem muitas dores de cabeça, mesmo porque, até aí, existe muito pouco perigo e quebra-cabeças, e creio que até os menores de "60 anos" chegarão lá tranqüilamente e se aventurarão por outras salas.

Você tem que ter sempre em mente que: todas as CHAVES serão necessárias e que os PRÊMIOS contam pontos e dão vidas.

O primeiro objetivo do jogo é conseguir o MAPA do castelo e, para isto, vamos mostrar toda a seqüência de movimentos necessários, tendo como orientação o mapa ampliado, na figura 4, das cinco salas envolvidas neste objetivo.

Iniciado o jogo na sala A7, faça o seguinte:

1 — Pegue a CHAVE amarela.

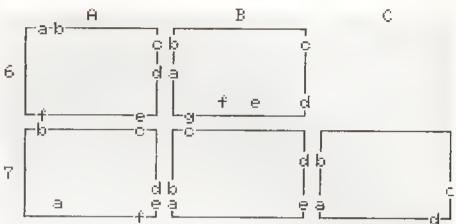



FIGURA 4

2 — Aguarde o SOLDADO chegar na PORTA A7a.

3 — Siga para a direita, abrindo a PORTA e posteriormente andando atrás do SOLDADO.

4 — No momento em que o SOLDADO der meia volta, salte e e suba para o piso de cima indo em direção à PORTA amarela.

5 — Pegue a CHAVE amarela com cuidado para não esbarrar nos ESPETOS do teto.

6 — Passe pela PORTA A7d e aparecerá a sala B7.

7 — Salte a falha do piso até chegar do outro lado da falha.

8 — Suba no TIJOLINHO e salte para pegar a CHAVE amarela.

9 — Empurre o TIJOLINHO para a esquerda até que ele fique com uma metade fora do piso. Se ele cair sem destruir o RATÃO, retorne à sala A7 e novamente volte a sala B7 pois, com isto, o TIJOLINHO voltará para o seu lugar de origem.

10 — Meça a altura do piso onde está o TIJOLINHO até o piso de baixo. Depois, meça esta mesma distância da ponta onde cairá o TIJOLINHO, para a direita e aguarde o RATÃO chegar aí, vindo da direita para a esquerda e, então... empurre o TIJOLINHO e "adeus RATÃO".

11 — Saia pela PORTA B7b e retorne à sala B7 para que o TIJOLINHO volte para o seu lugar de origem para dar passagem.

12 — Saia pela PORTA B7e e aparecerá a sala C7.

13 — Pegue a CHAVE azul clara e saia pela PORTA C7a retornando à sala B7.

14 — Suba ao piso de cima e empurre o tijolinho para perto da falha do piso.

15 — Suba no TIJOLINHO e salte para o piso do TRANSPORTE VERTICAL.

16 — Aguarde o momento certo, suba no TRANSPORTE VERTICAL e saia pela SAÍDA B7c e aparecerá a sala B6.

17 — Desça rapidamente do TRANSPORTE VERTICAL e abra a PORTA B6f.

18 — Empurre o JARRÃO para a direita (apenas dois passos), de tal forma que fique um espaço do tamanho do corpo do PRÍNCIPE entre o JARRÃO e a parede.

19 — Salte para cima, pegue o POTE VERDE (uma vida extra), dirija o PRÍNCIPE para a direita, posicionando-o sobre o JARRÃO com metade do seu corpo para fora do JARRÃO, à direita.

20 — Quando a BRUXINHA da direita chegar no extremo esquerdo do seu piso, salte para aquele piso, e, logo que chegar no chão daquele piso, salte novamente para pegar as duas CHAVES amarelas, deixe o PRÍNCIPE cair junto à parede e, imediatamente, salte de novo indo em direção à esquerda e, quando alcançar o final do piso desta BRUXINHA, deixe o PRÍNCIPE cair entre o JARRÃO e a parede.

21 — Empurre o JARRÃO para a esquerda e suba nele para auxiliá-lo a subir no piso da BRUXINHA da esquerda.

22 — Quando a BRUXINHA da esquerda chegar no extremo direito do seu piso, salte para aquele piso e, logo que chegar ao chão daquele piso, salte novamente para sair pela PORTA B6a e aparecerá a sala A6.

23 — Salte para o piso do TIJOLINHO e trate de destruir o REIZINHO do piso imediatamente abaixo, quando ele es-

FIGURA 5 — SALA C9 RESOLVIDA

tiver "passeando" na canto esquerdo.

24 — Salte para o piso onde estava o "recém-falecido" REIZINHO, aguarde a ONDINHA parar de ondular e, então, caia nela indo bem pelo direito e saia pela saída A6e e aparecerá a sala A7.

25 — Pegue o MAPA enquanto salta o SOLDADO, a CHAVE verde, a CHAVE amarela e, saltando o outro SOLDADO, saia desta sala pela ex-PORTA A7d e aparecerá a sala B7.

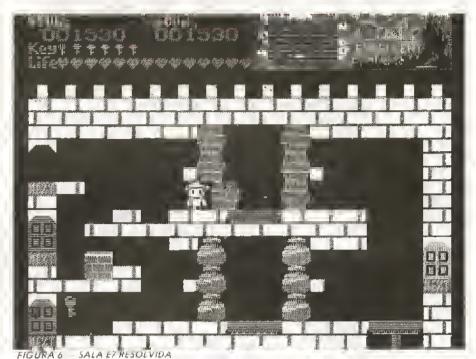
26 — Saia pela porta B7a e aparecerá a sala A7 novamente tendo-se, então, como única alternativa, "cair" na saída A7f.

Viu só como foi fácil??? Bem, agora, seu novo objetivo é colecionar cinco CHAVES verdes para poder pegar uma determinada CHAVE azul-escura e, para tal, a seguir, damos a seqüência de salas que você deverá percorrer para conseguir este segundo objetivo. Agora, você terá de escolher por qual das PORTAS ir para uma próxima sala, o quê e como fazer em cada uma delas.

FIGURA 6 — SALA E7 RESOLVIDA

1ª SEQUÊNCIA PARA CONSEGUIR A 1ª CHAVE LARANJA

A8 — B8 — C8 — B8 — A8 — B8 — C8 — C7 — B7 — B6 — C6 — D6 — D7 — E7 — D7 — D8 — E8 — D8 — D7 — E7 — D7 — D8 — E8 — E7 — D7 — D8 — E8 — D8 — D7 — E7 — D7 — D8 — D9 — C9 — D9 — E9 — D9 — D8 — C8 — D8 — D7 — C7 — B7 — A7 — A8 — A9 — B9 — C9 — D9 — E9

OBS.: Sugerimos que, antes de sair da sala, grave na fita a situação atual. É muito fácil você se enganar de saída e não poder mais voltar ou não ter mais chaves para as outras portas ou, simplesmente, se perder.

DICAS ESPECIAIS

• **Sala A8** — Destrua o RATÃO empurrando o SACO que está no piso de cima sobre ele e engane o FOGUINHO.

• **Sala A9** — Antes de pegar o OXIGÊNIO, treine como sair o mais rápido possível das ESTEIRAS, pois seu tempo é limitado.

• **Sala B8** — Empurrando o BARRIL no SOLDADO, ele é destruído e, destrua o REIZINHO, imprensando-o na parede com o auxílio do SACO que está no piso de baixo.

• **Sala B9** — Primeiro pegue a CHAVE da esquerda.

• **Sala C6** — Num salto, caia na ESTEIRA e, noutro, pegue a CHAVE enquanto o TIJOLINHO passa por baixo do PRÍNCIPE e use a falha do teto do piso do RATÃO para se esconder dele quando ele estiver vindo do direito para o esquerdo.

• **Sala C9** — Destrua o SACO que está no piso de cima (veja figura 5 da sala resolvida).

• **Sala D7** — Caia sobre os TRANSPORTES FLUTUANTES e caminhe normalmente (na mesma direção).

• **Sala D8** — Olho na mira. Destrua o SOLDADO do primeiro piso e, depois, use os JARRÕES como degraus para alcançar as PORTAS.

• **Sala D9** — Atraia os REIZINHOS para a ONDINHA e destrua todos eles.

• **Sala E7** — Destrua todos os SACOS e BARRIS (veja figura 6 da sala resolvida).

• **Sala E8** — Muita velocidade, raciocínio e exatidão (veja figura 7 da sala resolvida).

• **Sala E9** — Cuidado, os FOGUINHOS são "fogo".

FIGURA 7 — SALA E8 RESOLVIDA

Após a última parte do artigo, quando daremos as dicas para você conseguir libertar a PRINCESINHA, se você nos descrever o que acontece quando a PRINCESINHA é libertada (como é a tela que aparecerá) junto com as "senhas" de cada artigo e uma palavra-chave escondida nas revistas dos artigos, você receberá, com exclusividade, num diskete (não serão impressas na revista), os programas que foram usados para se extrair cada um destes desenhos que são operados pelo jogo como SCREEN1 e foram transformados em SCREEN2 para poderem ser impressos. A senha deste artigo é a palavra persistência. Colecione as revistas, pois as palavras-chaves necessárias só serão reveladas na edição da última parte do artigo.

Boa sorte e até o mês que vem.

PACLAND

1ª CASA

BREAK TIME

2ª CASA

3ª CASA

BREAK TIME

FAIRY LAND

1ª CASA

3ª CASA

4ª CASA

BREAK TIME
FAIRY LAND
BREAK TIME
FAIRY LAND
BREAK TIME
BREAK TIME
BREAK TIME
FAIRY LAND
2ª CASA

WAYNE ALLEN

BRUCE LEE
LANTERNA
FOGO
TREPADEIRA
ESCADA
S INÍCIO
FIM
NINJA

NEMESIS
O MELHOR PARA SEU MEX e MSX2 VOCÊ ENCONTRA NA NEMESIS
NEMESIS – PROGRAMAS UTILITÁRIOS
MSX-DOS TOOLS PLUS (LANÇAMENTO) ferramentas para auxílio na programação 20 BTNS
MSX HELLO! 1.0 multi-utilitário para uso com disk-drive 20 BTNS
MSX HARDCOPY 1.1 utilitário para impressão de gráficos 10 BTNS*
EASY GRAPH poderoso editor gráfico com recursos inéditos 30 BTNS
NEMESIS – PROGRAMAS APLICATIVOS
MALA DIRETA MSX 1.1 cadastro de clientes para 7.000 registros 30 BTNS
MSX-SAM VOICE SYNTHETIZER sintetizador de voz com 1 canal de som 15 BTNS*
MSX CHART 1.0 gráficos comerciais e estatísticos 20 BTNS
MSX PORTFOLIO 1.0 agenda eletrônica / lista telefônica 20 BTNS
I CHING horóscopo chinês no computador 10 BTNS*
TEXTO TOTAL 1.0 MTA (LANÇAMENTO) poderoso processador de textos para MTA 25 BTNS
TEXTO TOTAL 1.0 LADY (LANÇAMENTO) poderoso processador de textos com gráficos 25 BTNS
MSX TOP CAD (LANÇAMENTO) sensacional editor de projetos profissionais 40 BTNS
NEMESIS – DESK-TOP PUBLISHING NO MSX
MSX PAGE MAKER 1.4 editor de página com textos e gráficos 20 BTNS
MSX PAGE MAKER FONTES 1 22 diferentes letras para o PAGE MAKER 10 BTNS
MSX PAGE MAKER FONTES 2 22 diferentes letras para o PAGE MAKER 10 BTNS
MSX PAGE MAKER FONTES 3 22 diferentes letras para o PAGE MAKER 10 BTNS
MSX PAGE MAKER FONTES 4 22 diferentes letras para o PAGE MAKER 10 BTNS
MSX PAGE MAKER CARTOONS 1 diversas figuras para sua página gráfica 10 BTNS
MSX PAGE MAKER CARTOONS 2 diversas figuras para sua página gráfica 10 BTNS
MSX PAGE MAKER TITLES 1 alfabetos gigantes para títulos e destaques 10 BTNS
MSX PAGE MAKER SQUARES 1 diferentes molduras, adornos e vinhetas 10 BTNS
MSX PAGE MAKER KIT PAGE MAKER com todos os seus acessórios 90 BTNS
NEMESIS – JOGOS E PROGRAMAS EDUCATIVOS
O CONDE DE MONTE CRISTO aventura conversacional em português 15 BTNS
MEMPHIS aventura conversacional em português 15 BTNS
A GRUTA DE MAQUINÉ aventura conversacional em português 15 BTNS
AUTO KIT programa educativo para crianças 15 BTNS*
FARM KIT programa educativo para crianças 15 BTNS
PRACTICA – APLICATIVOS PROFISSIONAIS
DBASE II poderoso gerador de banco de dados 70 BTNS
SUPERCALC II poderosa planilha de cálculos 60 BTNS
CONTAS A PAGAR programado para uso com o DBASE II 20 BTNS
CONTROLE DE BANCO programado para uso com o DBASE II 20 BTNS
CONTROLE DE ESTOQUE programado para uso com o DBASE II 20 BTNS
PAULISOFT – APLICATIVOS E UTILITÁRIOS
AQUARELA super editor gráfico, 100% nacional 45 BTNS
FASTCOPY o copiador de discos mais rápido e seguro que existe 20 BTNS
EDTRONIC editor de circuitos eletrônicos 30 BTNS*
GRAPHIC VIEW editor vídeo-gráfico e de animação 30 BTNS
SPRITE MAKER editor de "sprites" com variados recursos 20 BTNS*
TOP CLI programa de cadastro de clientes 70 BTNS
BKP DISCO multi-utilitário para MSX com disk-drive 30 BTNS
MEGA MUSIC I ou MEGA MUSIC II músicas tiradas de jogos MEGARAM japoneses 11 BTNS
MSX TURBO acelera em até 60 vezes o BASIC MSX 30 BTNS
XSW – APLICATIVOS E UTILITÁRIOS
MSX EDARQ editor de arquivos em disco 40 BTNS
MSX VOX digitalizador de voz 100% nacional 35 BTNS
FLUXO DE CAIXA controle comercial de entradas a saídas 40 BTNS
M.P.O. SOFT VIDEO – VÍDEOS EDUCATIVOS EM VHS
DOMINANDO O MSX apresentação do MSX e seus periféricos 30 BTNS
CURSO DE BASIC 1 iniciação e programação basic no MSX 30 BTNS
CURSO DE DBASE II uma aula particular de DBASE II em vídeo 30 BTNS
SOFT-O-MATIC PROGRAMAS APLICATIVOS
BANCOS DE DADOS
MSX DATA BASE 1.1 fichário eletrônico fácil de usar 8 BTNS*
MSX DATA BAK 1.2 baco de dados com campos redefiníveis 8 BTNS*
MSX EASY DATA 1.0 cadastro redefinível fácil de usar 8 BTNS*
STOCK CONTROL 1.0 controle de estoques fácil de usar 15 BTNS*
STOCK CONTROL 2.0 controle de estoques profissional 20 BTNS
EDITORES GRÁFICOS
EDDY 1 editor gráfico fácil de usar 8 BTNS*
EDDY 2 editor gráfico com múltiplos recursos 8 BTNS*
CHEESE editor gráfico fácil de usar 8 BTNS*
GRAPHIC MASTER editor gráfico com "shapes" exclusivos 8 BTNS*
YAMAHA GRAPHIC ARTIST editor gráfico com variados recursos 8 BTNS*
AACKODRAW & PAINT poderoso editor gráfico com letras e texturas 8 BTNS*
T-PAINT 1.2 editor gráfico fácil de usar 8 BTNS*
THE DESIGNER'S PENCIL mais que um simples editor gráfico 10 BTNS*
THE MAGIC PAINT famoso editor gráfico do Apple agora para MSX 15 BTNS
EDITORES MUSICAIS
MUSIC STUDIO G7 poderoso editor musical com recursos inéditos 8 BTNS*
PSG MUSIC WRITER editor de música com ritmos variados 8 BTNS*
SUPER SINTH poderoso sintetizador de sons 8 BTNS*
THE MUSIC EDITOR editor de música de fácil manuseio 8 BTNS*
ELETRIC SOUND STUDIO poderoso sintetizador de sons e efeitos musicais 15 BTNS
PROCESSADORES DE TEXTOS
AACKOSCRIBE eficiente editor de textos com 40 ou 80 colunas 20 BTNS
THE BANK STREET WRITER processador de textos de fácil utilização 20 BTNS*
MSX WRITE 3.0 a nova versão do mais famoso editor para MSX 15 BTNS*
PRINT-X-PRESS II editor de "desk-top publishing" para seu MSX 20 BTNS
UTILITÁRIOS DIVERSOS
LETRASET 1.1 redefinidor de caracteres com versatilidade 8 BTNS
SUPER COPY 2.1 copiador de programas de fitas cassetes a disco 10 BTNS*
KNIGHT COMMAND 2.0 poderoso auxiliar para programação com MSX 10 BTNS*
D.O.S. HELP auxilia no uso do MSX com "disk-drive" 10 BTNS
DISK-IT facilita a operação do MSX com "disk-drive" 10 BTNS*
MSX DUAD 7.4 montador assembler / disassembler, editor, etc. 20 BTNS
PROGRAMAS DE USO GERAL
MSX GAME DESIGNER faça os seus próprios jogos com facilidade 12 BTNS*
STAR SEEKER excelente programa para amantes da astronomia 8 BTNS*
PRINT SEXY SHOP imprime mulheres nuas com sua impressora 8 BTNS
VÍDEO-GAMES MSX 1
CYBERBIG, BLACK TIGER 1, BLACK TIGER 2, BLOODY, CAPTAIN TRUENO 1, CAPTAIN TRUENO 2, TOY ACID GAME, COSMIC SHERIFF, MIKE GUNNER (LASER PISTOL), LIVINGSTONE II PART 1, LIVINGSTONE II PART 2, MECANO OASIS, XH-63 PATROL, ZANAC III, TENSION, WARLORDS, RANSACK, TALEF, HYPERTRONIC, RATH AE, SWING (NOVIDADE DA COMPILE), SABOTAGE, PASTEMAN e THE SMURFS.
CADA JOGO POR 1 BTN
5 JOGOS POR 4 BTNS
TODOS OS JOGOS (4 DISCOS) POR 20 BTNS
ATENÇÃO:
1 – O valor da BTN é equivalente ao do dia em que o pedido chega em nossas mãos;
2 – Os programas acima estão disponíveis em discos 5 1/4 e 3 1/2. Para 3 1/2 acrescente 4 BTNS por programa;
3 – Os utilitários MSX-DOS TOOLS I e o MSX HELLO! 1.0 não estão disponíveis em 3 1/2;
4 – Os programas assinalados com asterisco (*) podem ser gravados em FITA CASSETE;
5 – Consulte-nos sobre gravações de jogos em FITA CASSETE;
6 – Clube do leitor: 10% em todas as compras; 20% em compras acima de 70 BTNS e 30% em compras acima de 100 BTNS;
7 – O pedido mínimo é de 10 BTNS;
8 – Esta tabela está válida até o final de nossos estoques.
ENVIE VALE POSTAL OU CHEQUE NOMINAL A NEMESIS INFORMÁTICA LTDA.
CAIXA POSTAL 4583 CEP. 20.001 RIO DE JANEIRO – RJ
OU VENHA PESSOALMENTE AO "SHOW-ROOM" NEMESIS
RUA SETE DE SETEMBRO 92 SALA 1910 – CENTRO – RIO DE JANEIRO – RJ
TELEFONE: (021) 222-4900

ORIONSOFT
MSX
BANCO DE DADOS
MEGA ASS
MAXI DADOS
CARTUCHO
INTERFACE DIGITAL LEITORA DE FITAS
APLICATIVO
SIMULADORES
BOING 737